A
l33t Speak

The term "l33t Speak" (pronounced "leet") refers to a language or a notational system widely used by hackers. This notation is unique because it cannot be handwritten or spoken. It is an Internet-based notation that relies on the keyboard. It is simple to learn and has room for creativity. Web site [bbc 04] is just one of many online references to this topic.

Many other artificial languages or notational rules have been described or used in literature. The following are a few examples.

Elvish in J. R. R. Tolkien's *The Lord of the Rings*.
Newspeak in George Orwell's *Nineteen Eighty-Four*.
Ptydepe in Václav Havel's *The Memorandum*.
Nadsat in Anthony Burgess' *A Clockwork Orange*.
Marain in Iain M. Banks' *The Player of Games* and his other Culture novels.
Pravic in Ursula K. LeGuin's *The Dispossessed*.

The history of l33t speak is tied up with the Internet. In the early 1980s, as the Internet started to become popular, hackers became aware of themselves as a "species." They wanted a notation that will both identify them as hackers and will make it difficult for others to locate hacker Web sites and newsgroups on the Internet with a simple search. Since a keyboard is one of the chief tools used by a hacker, it is no wonder that the new notation developed from the keyboard. The initial, tentative steps in the development of l33t speak have simply replaced certain letters (mostly vowels) by digits with similar glyphs, so A was replaced by 4 and E was replaced by 3.

It was the development of sophisticated computer games in the early 1990s that boosted the popularity of l33t speak and prompted hackers to enrich it with features. Gamers started developing their own language, based on phrases heard in games, and hackers who played games (and there are many of them) naturally wrote such phrases in

l33t speak. An important example is the phrase "I am elite," whose l33t speak version is "1 4m 3l1t3." Hackers are notoriously bad spellers, so this phrase got first corrupted to "1 4m 3l33t," then mutated to "1 4m 133t," which gave l33t speak its current name. The final boost to the popularity of l33t speak was given by a very popular web comic called Megatokyo.

Like any other language or notational system, l33t speak has its grammar rules, but they are flexible, allowing users to be creative. The basic rules for replacing letters with digits and other keyboard characters are listed here, but new rules appear all the time and either become popular or are forgotten.

$$A \to 4,\ E \to 3,\ I \to 1,\ O \to \phi,\ O \to (),\ U \to |\_|,\ T \to 7,\ D \to |),\ W \to \backslash/\backslash/,\ S \to \$$$

(Notice the two versions of O.) Connoisseurs of l33t speak talk about classifying versions of this notation into classes or levels such as light l33t, medium 1337, hard |_337, and ultra $|\_33^-|^-$.

In l33t, `z` is used instead of `s` to construct the plural, `f` is generally changed to `ph`, and a short `u` is often replaced by the pair $\phi\phi$.

The term "digram" is used in English to indicate a pair of characters and l33t speak employs ϕd and ϕr to express the digrams "ed" and "er." For example, "1 4m 4 l33t h4xϕr" can be used instead of "1 4m 4 133t h4ck3r." (In ultra, this would be spelled "1 $\phi\backslash/\backslash/|\backslash|z\phi r$.")

Punctuation marks are sometimes omitted, and are rare in higher levels of l33t speak. Many chat programs allow the user to type only one sentence at a time, so there is no need for end-of-sentence periods. On the other hand, since l33t speak is commonly used to express surprise or pleasure, exclamation marks are popular and are sometimes repeated several times (a practice strictly prohibited by traditional copy editors). Low levels of l33t speak may use commas, but even those humble punctuation marks are omitted in the higher levels.

The vocabulary of l33t speak is mostly a corrupted form of English, but many phrases and spellings are unique to l33t speak. Here are some examples.

$\phi w|\backslash|$ or $\phi wn3d$. A popular l33t speak word. Its (very loose) meaning is "beaten" but it can also express awe, as in, "I ϕwn3d you" which means "I have beaten you good and proper", or ϕwn4ge!" which means "That was very nifty."

w$\phi\phi$t. This word, derived from "hoot," is interpreted to mean "yay," and is commonly used to express victory.

13wt. A misspelling of "loot" that came to mean a treasure, good merchandise, or possessions. Its most common use it to refer to pirated software, to items in a game, or promotional giveaways.

h4xϕr. The word for hacker or a skillful person. This is the most common occurrence of the ϕr digram.

ph33r. Fear. Most-commonly used in phrases such as "Ph33r m3!" or "Ph33r $|\backslash/|$y 1337 sk1llz!" It can also be spelled "ph34r."

sk1llz. A word derived from "skill" and referring to skill in some online activity such as programming or hacking. Often used in conjunction with "m4d." As a general rule, if one has sk1llz, one is to be ph33r3d.

m4d. Mad, commonly used as a descriptive term meaning great, for example, "h3s gϕt m4d sk1llz."

j$\phi\phi$. You, often used in phrases like "j$\phi\phi$ d34d f$\phi\phi$."

f$\phi\phi$. Fool, someone not bright or skillless.

> In any event, it's not a matter of liking or disliking, not a matter of skillful or skillless (how many Ls does that have?). No. When it comes to penguin shuffling, it's your patriotic duty.
>
> —Anonymous

jϕ. Yo, as in the greeting.

d$\phi\phi$d. Dude, used to address a colleague or an unknown person online.

suxϕr. Sucks, as in "7h1s suxϕr," an example of the ϕr digram.

l4m3r. Lamer, someone who is lame, an unfair person or someone who isn't fun to be with.

n$\phi\phi$b. Short for noobie, a corruption of newbie. Someone who is new to or is weak at something.

Usage of l33t speak is nonuniform. Some use it exclusively or almost so, while others frown on general use of l33t speak and insist that it should be used only in brief expressions, preferably expression of excitement.

◇ **Exercise A.1:** (No answer provided). Rewrite the following two l33t speak paragraphs in your language.

```
$4n DI3go- feDEr4L @gEN+s $t0rmEd tH3 HOU53$ Of tW0 $U5p3CTEd TeRrOr15+5 E@RLY
Y35+3rd@Y M0RniNg 4PHTer dEt3rMiNiNg theY WER3 P4Rt opH @ Pl0+ +0 De$TR0y u.5. C1+i3$.

jac0b M4R+Z 4nd cuR+iS hUgH3$, WELL knOWN 1n t3H Pc g4Ming C0MMUnI+Y 4$ "W@xX0R"
aND "L00d@|<r1$" resP3C+1vely, 4Re NOw 1N cU5+oDY 4ND @wA1+iN9 4 BA1l HeaRiN9.
```

Among my most prized possessions
are words that I have never spoken.

—Orson Rega Card

B
Virus Timeline

This timeline is meant to serve both as a historical survey and as a teaching tool. Most of the viruses described in this appendix have interesting features that make them unique or a first. Those are described here in some detail. Chapter 5 has several detailed descriptions of viruses and worms, including some of the ones mentioned here.

Several timelines of computer viruses can be found on the Internet. One reference is [IbmAntiVirus 05].

1949–50. First attempts to implement self-replicating programs.

1950s. An experimental game in which players use malicious programs to attack each other's computers is developed and used in Bell Labs.

1975. John Brunner publishes *The Shockwave Rider*, a science fiction novel in which computer "worms" spread across networks.

1981. Several of the first viruses seen "in the wild," (i.e., in the public domain) are found on the Apple II operating system, and are designated Apple viruses 1, 2, and 3. These viruses spread through Texas A&M University via pirated computer games.

1982. Another virus found on the Apple II computer and is designated Elk Cloner (the term *virus* was not used).

1983–84. Fred Cohen is the first to consider viruses a serious topic of scientific study and experimentation. He proposes a definition and conducts controlled experiments in virus propagation.

1986. Brain (Section 5.2), perhaps the first widespread virus (a BSI with stealth features), seems to have been written by two brothers in Pakistan who disseminate it on floppy disks with pirated software sold to tourists.

In December, a file infector named Virdem is introduced in Germany as a demonstration. It is quickly followed by the demonstration viruses Burger and Rush Hour.

1987. This is the first bad year and it signals the shape of things to come. In the fall, the Lehigh virus (Section 5.1), an early file infector, appears at Lehigh university in Bethlehem, Pennsylvania and infects `command.com` files.

The Christmas tree worm paralyses the IBM worldwide network.

In December, the Jerusalem virus appears at the Hebrew University of Israel. It was the first file infector that infected both `.com` and `.exe` executable files and also successfully modified interrupt handling routines so that it could reside in memory and be invoked frequently. A bug in this virus caused it to reinfect programs. The Jerusalem virus was preceded by three variants designated Suriv 1, 2, and 3 (suriv is virus spelled backward) and may have been deployed by the same author.

Two more viruses appear later in the year, the stoned virus (the first master boot record, or MBR infector), apparently written by a student in New Zealand, and the Vienna virus, written by an Austrian high school student. The latter is completely disassembled and analyzed, and its code published.

Last (and perhaps least), a virus appears in South Africa that deletes files on Friday the 13th.

Origins of Friday the 13th

Many stories, anecdotes, and beliefs explain why Friday the 13th is considered by many an unlucky day. Perhaps the most important of those is a historical event. On Friday, October 13, 1307, the Pope of the Roman Catholic church, together with the King of France, sentenced the "Knights Templars" to death and ordered the torture and crucifixion of their leader.

Traditional beliefs have it that Eve tempted Adam with the apple on a Friday, the Biblical Flood, the confusion at the Tower of Babel, and the death of Jesus Christ all took place on a Friday. Also, 13 was the number at the Last Supper following which Judas betrayed Jesus.

The Belgian writer Georges Simenon was born a little after one AM on Friday, 13 February 1903. Being superstitious, his mother Henriette had the birth date officially falsified and recorded as February 12th.

The fear of the number 13 is known scientifically as *tridecaphobia,* and is perhaps the most common of all superstitions.

1988. The Internet worm (Section 3.4) spreads through the United States DARPA network by exploiting security weaknesses in the `finger` and `sendmail` UNIX utilities. In a rare stroke of luck, its author is promptly identified, tried, and punished.

The first good virus (or anti-virus virus) is released. Its task is to detect and remove the Brain virus. There are two versions of this virus, written by Denny Yanuar Ramdhani in Bandung, Indonesia and named the Den Zuk viruses

This year also sees another innovation, a self-encrypting virus. First found in Germany, the cascade virus is a file infector that encrypts itself with a random key.

The ping-pong virus (also known as "bouncing ball" or "Italian") appears at the university of Turin in Italy in March. It becomes the most common and best known boot sector virus (BSI) and keeps this title for a while. This virus had a small bug that caused it to crash computers based on the Intel 80286 microprocessor and its successors, which made the ping-pong virus easy to detect.

Finally, after several years of attacks, infections, and much damage inflicted by viruses, the topic of malicious software starts attracting the attention of the media. Newspapers and magazines publish articles and news briefs about occurrences of viruses and worms, speculations as to their origins, and descriptions provided by virus detectives.

1989. The `Dark Avenger.1800` virus is unleashed from Sophia, Bulgaria in January. It is named Dark Avenger after its anonymous creator, and it represents the next step in virus sophistication (some might say, the next generation of viruses). It spreads fast because it infects executable files as they are opened, even if they do not execute. Also, its payload is dangerous. It performs slow data diddling to files on the disk, so that when its damage is finally discovered, even files backed up weeks before are already corrupted.

◇ **Exercise B.1:** The `Dark Avenger.1800` virus infected executable files as they were opened, even if they did not execute. Why does this make it a fast-spreading virus?

In October, the "Frodo lives" virus emerges from Israel. This is an advanced stealth file infector. It saves the original length of every file infected and displays the lengths when the user asks for a directory listing. It also tries to intercept attempts to read infected files and it sends the original, clean files instead. This virus triggers on September 22 of any year (if it happens to execute on that date), when it displays the message "Frodo lives" and tries (unsuccessfully, because of a bug in its code) to install a Trojan horse. (Frodo Baggins is one of the main characters in the novel *The Lord of the Rings* by J. R. R. Tolkien.)

1990. The flip virus (a slow file infector also designated flip-2343 because it increases the size of infected files by this number of bytes) escapes from Switzerland and is seen in the wild everywhere. This is perhaps the first successful multipartite virus (both BSI and file infector, Section 2.9) and is also polymorphic (appears as different bit strings, Section 2.21), which delays its identification, isolation, and successful removal. Flip is a slow infector because the only way for it to infect a computer is when a flip-infected file is executed. The file can come from an external disk, can be downloaded from a network, or be an email attachment.

Flip got its name because its payload is to flip horizontally the display on the monitor screen (only EGA or VGA monitors) on the second day of each month, between 16:00 and 16:59. Other flip viruses appear in future years, with other payloads in addition to the relatively harmless flipping.

Symantec Inc., already a recognized company in the field of computer security, launches Norton AntiVirus, one of the oldest anti-virus programs. At the time of this writing,

the program still bears the same name and is regularly updated (both the software and the virus definitions).

1991. The tequila virus is a the first widespread polymorph of flip, probably generated by the same person who wrote the original flip virus.

The Bulgarian virus developer Dark Avenger announces in March that he is working on a new, dangerous virus that can mutate in billions of ways. This threat will materialize in 1992 (see MtE).

The first virus kits appear. Version 1 of VCS (virus construction set) appears in March in Hamburg, Germany. It was written by the Verband Deutscher Virenliebhaber (community of German virus lovers). It is followed by VCL (virus construction lab, implemented by Nowhere Man) and in August by PS-MPC.

VCL is an attempt by a virus writer calling himself Nowhere Man to create a user-friendly package that will allow inexperienced programmers to create their own viruses. VCL has menus for the infection type, encryption, and payload, allowing a would-be virus creator to easily generate code in either assembler language or directly as an executable `.com` file. PS-MPC is a virus code generation file (PS stands for Phalcon/Skism, presumably indicating the anonymous writers) in August.

Virus workers claim that there are upward of 1000 viruses in existence.

1992. Dark Avenger finally releases his long-promised mutation engine (dubbed MtE or DAME). It turns out to be a toolkit that converts ordinary viruses into polymorphic ones.

It takes a while for anti-virus software to get to the point where viruses that employ MtE can be detected. In future years, MtE continues to be a source of inspiration for those planning to implement and unleash polymorphic engines.

> The demo virus which accompanied the [mutation] engine contained the text: "We dedicate this little virus to Sara [sic] Gordon, who wanted to have a virus named after her."
>
> —From [avenger 05].

This year marks the first worldwide panic about a computer virus. The Michelangelo virus (Section 5.3) is touted in the media as a global threat, but turns out to infect very few computers.

Statistics: There are now 1300 viruses in the wild (although many appear to be dead). In response, more and more computer users purchase anti-virus software, encouraging more software makers to jump on this bandwagon (many later alight).

1994. The first major virus hoax, Good Times, appears (hoaxes are discussed in Section 6.5). It warns about a destructive virus that erases an entire disk drive if an email message with the subject "Good Times" is opened. Rumors persist for months, and then resurface in future years.

A destructive, polymorphic virus called Pathogen (or alternatively, SMEG) appears in England. Analysis shows that Pathogen is really two sibling viruses, `SMEG.Pathogen`

and `SMEG.Queeg`. The virus author, who calls himself the black baron, claims to have written them in a language he calls the Simulated Metamorphic Encryption Generator (SMEG). The two siblings are highly polymorphic, and mutate to become completely different bit strings in each infection.

In a rare example of successful police work, the author is tracked down by New Scotland Yard's Computer Crime Unit. He is identified as Christopher Pile, is tried in November, and is sentenced to 18 months under the Computer Misuse Act of the United Kingdom.

> Ay! Then, Miss Newson, ye had better say nothing about this hoax, and take no heed of it. And if the person should say anything to you, be civil to him or her, as if you did not mind it—so you'll take the clever person's laugh away.
>
> —Thomas Hardy, *The Mayor of Casterbridge*, 1885.

1995. This is the year of the macro virus (Section 2.10). The first such virus, Concept, is discovered in May. It is written in WordBASIC (an interpreted programming language similar to Visual Basic for Applications and executed by Microsoft Word). Concept infects documents of Word versions 6 and 7 on any computer platforms (Word is supported on Windows and Macintosh). It seems that the only payload of Concept is to display the message "`REM That's enough to prove my point.`" Anti-virus software makers are not prepared for a macro virus. After analyzing Concept they assess it as a weak infector, but in fact it becomes one of the most prevalent viruses in the mid 1990s.

While the good guys try to come to terms with the new macro virus, virus developers decide they like this approach and they write and release a few more macro viruses this year.

1996. In January, the Boza virus (sometimes misspelled baza) is discovered. Boza is the first virus to spread only under the Microsoft Windows 95 operating system. Even though it has been seen in many geographic locations, it is not considered a serious threat to Windows 95 users. Boza was written by the Australian virus group VLAD and is named after a text string that it contains `Please note: the name of this virus is [Bizatch] written by Quantum/VLAD`.

Two variants, `Boza.B` and `Boza.C` are later released, probably in attempts to correct bugs in the original Boza, but have no noticeable effect.

In July, the anti-virus community learns of a new nondestructive macro virus (promptly named Laroux) that infects Microsoft Excel files (files with an `.xls` extension) for Excel versions 5 and 7 running under Windows 3, Windows 95 and Windows NT, but not on the Macintosh. Once an infected Excel document is opened, the virus will be active every time the Excel program is run, and will infect any workbook that's created or opened. Laroux was written in Visual Basic for Applications (VBA), a macro language based on Visual Basic from Microsoft.

Laroux consists of two macros, `auto_open` and `check_files`. The former is expanded whenever an infected Spreadsheet is opened, followed by the latter macro which determines the startup path of Excel. The virus creates a file titled `personal.xls` with a module called `laroux`, hence its name.

Laroux is one of the most common viruses, but fortunately it has no payload. It just replicates.

Staog is the first Linux virus, discovered in the Fall. It is written in assembler and it infects only Elf-style executable files in the Linux operating system. It copies itself into memory and tries to infect Elf-style executables when they are executed. Staog exploits three known vulnerabilities (mount buffer overflow, tip buffer overflow, and one suidperl bug) in Linux in an attempt to gain superuser status.

Staog is named after the text string "Staog by Quantum/VLAD" that was discovered in it. VLAD is the name of an Australian virus group that also wrote the first Windows 95 virus, Boza

1997. More hoaxes abound. The following hoax about five viruses is quoted from "Computer Crime: An Emerging Challenge for Law Enforcement," an article by the two PhDs David L. Carter and Andra J. Katz. It was published in the December 1996 edition of the FBI's Law Enforcement Bulletin.

> Virus Introduction
>
> Computer viruses, created for a variety of reasons, can have many different effects, depending on the creator's intent. To illustrate, several new insidious viruses have been found.
>
> "Gingrich" randomly converts word processing files into legalese often found in contracts. Victims can combat this virus by typing their names at the bottom of infected files, thereby signing them, as if signing a contract.
>
> "Clipper" scrambles all the data on a hard drive, rendering it useless.
>
> "Lecture" deliberately formats the hard drive, destroying all data, then scolds the user for not catching it.
>
> "Clinton" is designed to infect programs, but it eradicates itself when it cannot decide which program to infect.
>
> "SPA" examines programs on the hard disk to determine whether they are properly licensed. If the virus detects illegally copied software, it seizes the computer's modem, automatically dials 911, and asks for help.

1998. In June, the CIH virus, also known as Chernobyl, is discovered in Taiwan. Local authorities point to Chen Ing-hau as the writer of this virus, which derives its name from his initials.

The payload of this virus will first be triggered on April 26, 1999, causing many computer users to lose their data. The total loss is estimated in the hundreds of millions of dollars.

CIH searches for empty, unused spaces in executable files it attempts to infect. On finding such spaces, it breaks itself up into smaller pieces and inserts its code into them. To disinfect a file infected by CIH, anti-virus software looks for these small viral pieces and removes them from the file.

In August, the StrangeBrew virus rears its harmless head. This is the first virus to infect Java files. It can spread from a Java applet or a Java application to another, but only if executed locally, not over the Internet.

The virus searches for existing `.class` files and modifies them to append a copy of itself to the file and include a call to the virus' code in the first instruction. When such a file is later executed, the virus is executed first. The infector routine in StrangeBrew has bugs, as a result of which it rarely infects files in its host correctly. Most of the time it crashes the host when it attempts an infection.

The StrangeBrew virus is based on Java, which makes it capable of executing on virtually any platform that can run Java programs (all Windows and Linux platforms as well as PDA devices that have Java runtime installed).

StrangeBrew does not inflict any damage; it just spreads itself.

1999. March marks the first appearance of the Melissa menace (Section 5.5). The virus (with the official name `W97M_Melissa`) originates in an Internet `alt.sex` newsgroup. This is a macro virus that attacks the Microsoft Word 97 and Word 2000 applications and propagates via email attachments. Melissa executes a macro in a document attached to an email message, and this macro locates the owner's Outlook address book and forwards the document to 50 of the addresses found there. This technique is the reason for its unusually fast spreading. The virus also infects other Word documents and subsequently mails them out as attachments.

At the time of writing, Melissa and variants are still seen in the wild and the very latest about this threat can be found in [melissavirus 05].

An important first this year is the BubbleBoy worm. As soon as email users got used to the idea that email attachments can be dangerous, along comes this worm and teaches them another lesson. Merely *opening* an email message can infect the computer. The worm exploits a security weakness in Internet Explorer 5 (IE, a common Web browser, but the virus affects only IE installations that have Windows Scripting Host) and the fact that Microsoft Outlook, a popular email program, automatically opens email messages in a lower window in the program.

This worm spreads fast because it locates all the Outlook and Outlook Express e-mail address books in the computer and emails itself to every addressee in them. Fortunately, BubbleBoy is relatively harmless. It modifies the owner's email settings by changing the owner's name to BubbleBoy and the organization's name to Vandelay Industries. These are fictitious names taken from a popular television program.

Perhaps the worst feature of this worm is its successful spread, which may tempt other miscreants to come up with similar, but more destructive, worms in the future.

Another first for this year is the tristate macro virus (official name `O97M/Tristate`) and its many variants. This virus is written in Visual Basic for applications (VBA) and its name implies that it infects documents for Microsoft Word, Excel, and PowerPoint, three components of the MS Office 97 suite of applications. An unusual feature of this virus is the large number (at least 11) of its variants.

The virus removes all the macros from the MS Word global template. This, and the fact that it infects the three applications, is its only payload.

2000. This is the year of Love Bug, also known as the ILOVEYOU virus. First appearing in early May, this virus is the most "successful" email virus to date and is destructive. Within hours of its release it spreads to every continent and infects tens of thousands of computers. The number of machines infected after one day is estimated at 45 million. Its fast spread is attributed to the fact that many users save large numbers of old email messages and also have large address books with many correspondents. The virus can find all these addresses and it automatically generates email messages to all of them (this author has also received one from a student, but was unaffected because he uses a Macintosh and doesn't use Outlook).

This virus arrives as a Visual Basic script attachment in an email message whose subject line claims "I love you." It deletes audio, video, and image files. It also locates usernames and passwords and sends them to its author.

A suspect, 23-year old Reomel Ramones, is located and arrested within a week in Manila, the Philippines. Police charge him with being the originator of this virus, but his relatives blame his girlfriend's sister of creating the virus.

> One of the nastiest I've seen.
>
> This worm spreads at an amazing speed.
>
> It began spreading like wildfire, taking out computers left, right and centre.
>
> It's a particularly malicious virus.
>
> It is compromising security and confidentiality.
>
> It can go into private e-mails and forward them to anybody in your contacts book.
>
> I was looking for some deeper meaning in the last two major virus assaults. Each one has seven letters and three vowels, and if you rearrange the letters, MELISSA and LOVE BUG spell: BIG VOLUME SALES.
>
> —Experts' comments on the Love Bug virus.

This year the title "a first" belongs to the Stages worm, the first malware that infects text files. The worm enters a computer as an email attachment named `Life_stages.txt.shs` (`.shs` is the extension of Microsoft Scrap Object files. These files are executable and can contain many different types of objects), but the `.shs` extension is not displayed by the Windows operating system. When the attachment is opened, it is readable (it seems to joke about the male and female stages of life), but a script is simultaneously running in the background, infecting and deleting files. The worm locates addresses in the address books of Outlook, ICQ, mIRC, and PIRCH, and then mails itself as an attachment to all the addressees found. Thus, email users can no longer assume that text attachments to messages are safe.

On Monday, February 7, large, well-aimed and well-planned distributed denial-of-service (DDoS) attacks against Yahoo, eBay, Amazon, and other popular Web sites knock them

offline for several hours. What is especially frightening about this attack is that it comes from many servers; it is distributed.

2001. The nimda virus/worm is discovered on September 18. This virus (which affects Windows 95, 98, Me, NT 4, and 2000) has a worm component that spreads by sending email messages with an attachment called `readme.exe`. Nimda (whose name is the reverse of "admin") is a first in two areas (1) it infects files in Web sites and (2) it employs zombies to scan for vulnerable sites. The affected Web sites send the infected files to anyone downloading the files, and the zombies make it possible for nimda to reach Web sites located behind firewalls.

Nimda employs as many as five different methods of replicating and infecting computers, which makes it one of the most sophisticated viruses to date.

Once Nimda infects a computer, it proceeds in four steps as follows:

• Infection. Nimda locates `.exe` files in the computer and infects them. The infected files spread the infection when they are exchanged between computers.

• Mass mailing. Nimda locates email addresses in the address book of the computer's email client. It then searches local HTML files for more addresses. When done, it sends a message with the `readme.exe` attachment to each addressee.

• Web worm. Nimda scans the Internet in an attempt to locate Web servers. Once a server is found, the worm tries to infect it by exploiting several known security holes. If this succeeds, the worm selects Web pages at random on the site and infects them. Visitors downloading these files and executing them will get infected by the virus.

• LAN propagation. The worm component searches for file shares in the local network, either from file servers or from user computers. Once file sharing is found, the worm places an invisible file in any directory that has DOC and EML files. When users later try to open DOC or EML files from these directories, Word, Wordpad, or Outlook will execute the invisible file, thereby infecting the computer.

On February 12, the infamous "Anna Kournikova" worm (formal name `VBS.SST@mm`) starts its rounds. It arrives as an attachment named `AnnaKournikova.jpg.vbs`. When a hapless email reader clicks it, the worm searches the Microsoft Outlook address book and emails itself to every address found in it. On January 26, the worm attempts to direct the computer's Web browser to an Internet address in The Netherlands, which may point detectives looking for its perpetrator in the right direction. An important feature of this worm is that it's written with a virus kit, a tool that makes it easy for beginners to write rogue software. The messages sent by this worm have the subject line "`Here you have, ;o)`" and the message body "`Hi: Check This!`".

In one of those rare success stories that warms everyone's heart, the creator of this virus, 20-year old Jan de Wit of Sneek, The Netherlands, gives himself up, is tried and sentenced, but only to 150 hours community service because he is a first-time offender.

The verdict stated that de Wit "was not a layman in the field of computer viruses. He works in a computer store and collected viruses—about 7,200, according to himself. [The collection was confiscated.] The defendant must have been very aware of the

consequences of his acts. The virus he spread was a hindrance, causing worry and annoyance among Internet users worldwide."

Three worms, Sircam, CodeRed, and BadTrans, create headaches for virus workers (and revenues for anti-virus companies).

Sircam spreads as attachments to email messages sent through Windows Network shares. A typical message has one of many sender names and subject lines, but the message body is either `Hi! How are you? I send you this file in order to have your advice`, or something similar. The main innovation of this virus is the way it modifies the default EXE file startup registry key from `HKCR\exefile\shell\open\command` to `""[windows_drive]\recycled\SirC32.exe" "%1" %*"`. This results in an activation of the worm (from its location in folder `recycled`) whenever an `.exe` file is launched; ingenious!

Sircam has two payloads, but because of a bug neither of them works. The first payload is to delete all the files from the startup drive (this occurs on October 16 and in one of 20 cases). The second payload is to create a file and append text to it until it fills up the entire drive. This text contains the string `SirCam`, which is the reason for the particular name of this worm.

In July, CodeRed appears and expands at a terrific rate, much faster than any worm preceding it, infecting approximately 360,000 hosts in its first twelve hours of activity. This worm spreads by exploiting a security hole in the popular Microsoft Internet Information Server (IIS) software. Once it infects a server, it starts scanning the Internet in an attempt to locate other vulnerable servers. Once a month, the worm becomes active. For a few hours it only spreads, then it starts a Denial-of-Service (DoS) attack against `www1.whitehouse.gov` (the White House Web site), and finally it goes back to sleep.

A variant appears in 2002.

In April, the BadTrans worm (formal name `W95/Badtrans.B@mm`) is brought to life. This is a worm that spreads in attachments to email messages sent from computers running any 32-bit version of Windows. Once an email user clicks on the attachment, the worm executes. It places three files in the computer that act as an email worm and a Trojan horse. The worm component spreads itself by automatically sending infected answers to all the unread email messages in the user's inbox. The Trojan horse (file `HKK32.EXE`) is a variant of an older Trojan that steals passwords. It sends all the information it obtains to email address `ld8dl1@mailandnews.com`.

Lots of activity for one year!

2002. January marks the birth of `SWF/LFM-926`, the first virus to infect Macromedia Flash files (those with extension `.swf`). Flash is a popular program to display animation and special graphics effects. These files can execute scripts, a feature that makes it easy to develop complex animation, but is now found to be a security trap. As it infects `.swf` files, the virus displays the message `Loading.Flash.Movie`, which contributes the `LFM` in its name.

David L Smith, the creator of Melissa (Section 5.5), is sentenced to 20 months in prison.

The success of the Anna Kournikova worm in 2001 has encouraged hackers to continue with worms using celebrity names as a social engineering technique to lure victims, and at least three worms thus named are released in 2002.

• Shakira worm. Released in June, this worm (formal name `VBSWG.AQ`, where `VB` stands for Visual Basic) starts spreading through mIRC chats and email messages sent through Outlook. The subject line is `Shakira's Pictures` and the message body is `Hi: i have sent the photos via attachment, have funn...` The infected attachment is a file titled `ShakiraPics.jpg.vbs`.

This worm is written in Visual Basic Script and was generated with the `VBSWG` virus kit.

• Britney Spears is a very similar worm (appears in March) with the subject line `RE:Britney Pics`, message text `Take a look at these pics...`, and infected attachment `BRITNEY.CHM`.

• Jennifer Lopez worm (named `Loveletter.CN` with `VBS.Lopez.A@mm` as an alias) appears to have been written in Algeria. It arrives in an email with the subject line `Where are you?`, a message body `This is my pic in the beach`, and an infected attached file titled `JENNIFERLOPEZ_NAKED.JPG.VBS`. As part of its payload, this worm places in the Windows registry a key that causes it to execute each time Windows is started.

The Klez worm (actually, a worm/virus combination, dubbed `W95/Klez@mm`) arrives in October, probably from Asia, perhaps from China. Like many other worms, it enters the computer in an email message. It places in the computer a polymorphic `.exe` virus called `ElKern`. The Klez worm employs a variety of subject lines, such as `Hi`, `Hello`, `How are you?`, `Can you help me?`, `We want peace`, `Where will you go?`

As part of its payload, Klez removes autostarting registry keys of security and anti-virus software. As a result, this software or parts of it are disabled next time Windows starts. The virus also stops many processes and corrupts many files, most notably anti-virus checksum files and integrity checker databases. This worm/virus has several variants.

The month of September sees the arrival of the Bugbear worm (`W32.Bugbear@mm`). It attempts to place a keystroke logger and a backdoor in the computer and tries to terminate the processes of various antivirus and firewall programs.

The backdoor installed by Bugbear opens port 36794 and waits for commands from its author. The commands can order the worm to perform several actions as follows:

• Copy files.

• List files and deliver the list to the hacker.

• Delete files.

• Start processes.

• Terminate processes.

• List processes and deliver the list to the author.

• Deliver saved keystrokes to the owner in encrypted form.

• Deliver the following items of information to the owner: (1) Username. (2) Type of processor. (3) Version and build number of Windows. (4) Memory size and availability. (5) Types and physical characteristics of input/output volumes. (6) Network resources and their types.

2003. The slammer worm (`W32.Slammer`, alias Sapphire) appears out of the blue in January. This menace is different from the run-of-the-mill worm because it infects only computers running Microsoft SQL Server 2000 or MSDE 2000, i.e., servers. It uses UDP port 1434 to exploit a buffer overflow weakness in MS SQL servers. End-user machines are not affected. Another uncommon feature is that slammer does not write itself to the disk; it stays in memory until the computer is restarted (but if the computer hasn't been patched against slammer, it is likely to catch the worm again). However, the worm generates a massive amount of data packets, affecting Internet traffic all over the world.

The Blaster worm (`W32.Blaster.C.Worm`) surfaces in August. It exploits a certain vulnerability (with the technical name of DCOM RPC, described in Microsoft Security Bulletin MS03-026) and uses TCP port 135 to target computers running Windows 2000 and XP. In contrast to most other worms, Blaster does not search for email addresses and doesn't mail itself en masse.

Blaster is triggered by the following complex timing condition. From January to July it is triggered every day from the 16th until the end of the month. From 16 August until 31 December, it is triggered every day. The payload is a Denial of Service (DoS) attack on `www.windowsupdate.com`. The obvious aim is to prevent victims from downloading a security patch from Microsoft.

Another fast-spreading worm this year is sobig (`W32.Sobig.F@mm`). This is another mass-mailing worm that looks at many files (more precisely, files with extensions `.dbx`, `.eml`, `.hlp`, `.htm`, `.html`, `.mht`, `.wab`, and `.txt`) for addresses and mails itself to every address. It also spoofs the sender's address in these messages, using addresses found in the victim's computer. Thus, this worm becomes a source of spam. An excellent review of this malware can be found in [Skoudis 05].

Sobig deactivates itself (an unusual feature) on 10 September 2003. However, if the computer clock is out of date, it (the computer) may contribute to the worm's spread past the deactivation date.

Experts estimate that the Blaster and Sobig worms have turned August 2003 into the worst month ever for virus incidents. Obviously, things are getting worse.

2004. January. The MyDoom threat is unleashed. Known as either mydoom or novarg (`W32.Mydoom.A@mm` or `W32.Novarg.A`), this is an email worm that carries an infected attachment with one of the extensions `.bat`, `.cmd`, `.exe`, `.pif`, `.scr`, or `.zip`. Mydoom becomes the most widely-spread worm to date. It is estimated that at its peak, one quarter of all email messages carried this menace.

Mydoom is a sophisticated worm that installs a backdoor by opening TCP ports 3127 through 3198, through which the worm's owner can connect to the computer and to its network resources. The owner can also send any files to the affected computer through

these ports. Several pieces of rogue software, among them doomjuice, deadhat, and mitglieder, infect computers through this method.

The worm is triggered on 1 February 2004. There is a 25% chance that the worm will start a DoS attack on that date and continue this until 12 February 2004. If this happens, the worm does not mail itself from the infected computer, but the backdoor stays in the computer indefinitely.

The DoS attack is aimed at the SCO Group, a company that tried to sue several entities for illegally using an open-source version of its UNIX programming language. SCO offers a $250,000 reward to anyone helping in the arrest and conviction of Mydoom's originator(s).

January. The bagle worm (`W32/Bagle-mm`) starts spreading. This is a typical worm that arrives as an email attachment, scans the computer for email addresses and sends itself to all the addresses found. It is sent as a message with a subject line `Hi`, a body that includes the words `Test, yep`, and an attachment with extension `.exe`. Bagle floods computer networks all over the world, but does not have any other destructive payload. Several variants appear in 2005.

March. The Netsky worm (`w32.netsky.d@mm`) pops up. This version and its variant `W32.Netsky.C@mm` are mass-mailing worms. They scan drives `C` through `Z` of a PC for email addresses and email themselves with an infected attachment to all addresses found. The subject, body, and attachment names are selected at random from a set of names, except that the attachment has the extension `.pif`.

In late March, the Witty worm is unleashed to infect Macintosh computers. The worm exploits a vulnerability in BlackICE/RealSecure, firewall software from Internet Security Systems [ISS 05]. The vulnerability is discovered on 8 March with a patch issued by ISS the following day. The details of the vulnerability are published by eEye Digital Security on 18th March, and the worm appears about three days later. It infects every vulnerable Macintosh (about 12,000 computers) within 45 minutes (which translates to about 4.45 computers infected each second).

The witty worm is small, less than 700 bytes, which enables it to send a copy of itself in a single ethernet packet. Once arriving at a computer, it repeats the following two steps: (1) It attempts to replicate itself by generating 20,000 such packets and sending them to random IP addresses with random ports. (2) It locates a point on the hard disk at random and rewrites 65 Kb of data. After several repetitions, the artificial data written by the worm causes a freeze or a crash, and the computer has to be restarted.

Many viruses and worms fail in their destructive mission because of bugs in their code, but the Witty worm is bug free. This implies that it was written by an expert, and from scratch, not from a virus kit. This expert knows how to write Macintosh programs, is willing to wait for the right moment when a vulnerability is discovered, and doesn't mind if his creation infects only a small number of machines. These sad but true conclusions are summarized here.

- Some malware writers are experts, not bored teenagers.

• A worm can be fast propagating and also destructive, not being satisfied with just launching a DoS attack.

• Anti-virus software is useful and important, but it cannot identify new, as-yet-unrecognized malware and therefore does not guarantee a clean computer.

• There will always be users who ignore news and messages about new security patches, or are just too lazy to install them.

• Macintosh computers are relatively, but not absolutely, safe from malware.

May. This is the month of the sasser worm (`W32.Sasser.Worm`). This is a fast-spreading worm that exploits the MS04-011 (LSASS) vulnerability, a security weakness caused by a buffer overrun in the Local Security Authority Subsystem Service (LSASS). This becomes the major security hole of 2004. Because of it, sasser enters a computer through this vulnerability and not as an email attachment. Copy cats hear of the LSASS hole and immediately release a stream of rogue worms with names such as Korgo, Bobax, Cycle, Kibuv, and Plexus.

Upon infecting a computer, sasser starts 128 scanning threads that generate random IP addresses in an attempt to find vulnerable computers. Computers are probed on port 445 which is the default port for Windows SMB communication on NT-based PCs.

Sasser affects computers running Windows XP or Windows 2000 that are connected to the Internet without a firewall. A security patch is quickly issued by Microsoft.

It seems that the only damage this worm inflicts is crashing the computer (probably because of a bug in its code). An 18-year-old German high school student confesses to being the author of the worm. He's suspected of releasing another version of sasser.

A first in 2004 is malicious software that infects cell telephones running the Symbian operating system. Examples of such worms are `Toquimos.A`, `Skulls.A`, and the Cabir family (see year 2005).

Another first in 2004 is vulnerable jpeg images (Section 2.11). Normally, an image file has no executable code, and so cannot be infected. However, Microsoft has a software product that displays such images, and it had a security flaw in the form of buffer overrun. This flaw makes it possible, at least in principle, to construct a jpeg image that when viewed with this software will install a malicious program that can take over the computer and convert it to a zombie. Two malicious programs that take advantage of this flaw appear almost immediately. They are dubbed JPGDownloader and JPGTrojan. Microsoft very quickly issues a security patch to fix this buffer overrun.

2005. A Bagle variant `Bagle.AY` is found in January. Like its older relative `Bagle.AX` the new variant is polymorphic and arrives in email with randomly selected subject and attachment. It also has Peer-to-Peer spreading capabilities and contains a backdoor that waits for commands on TCP port 81. It is programmed to cease its activity on 25 April, 2006.

A new variant of the MyDoom worm, `MyDoom.AI` appears in January and uses social engineering to entice readers to open attachments. It arrives in email messages with infected `exe`, `scr`, `pif`, or `zip` attachments. Some messages contain sexually explicit images and claim that the attachment contains passwords for adult Websites.

A new cell telephone virus, lasco (alias `SymbOS/Lasco.A` and `EPOC/Lasco.A`), also appears in January and infects mobile telephones that run the Symbian operating system, support bluetooth, and are in discoverable mode (see also page 42).

This virus replicates over bluetooth connections and arrives in the message inbox of the telephone hidden inside a file called `velasco.sis`. When the user clicks on this file and agrees to install it, the virus is invoked. It immediately starts looking for new telephones to infect over bluetooth. It also inserts itself into other `sis` files it finds in the telephone. If such infected files are later copied into another telephone, the virus installer will be invoked with the first installation task, and ask the user to accept the installation of Velasco.

Cabir (aliases `SymbOS/Cabir.A`, `EPOC/Cabir.A`, `Worm.Symbian.Cabir.a`, and `Caribe virus`) also infects mobile telephones by exploiting the same vulnerability and is generally very similar to lasco.

A first this year is the `Duts.1520` virus (aliases `WinCE/Duts.1520`, `WinCE.Duts`, and `Dust`) a file infector that attacks the PocketPC platform. Duts affects ARM-based devices only. This is a short program (1520 bytes), apparently written in assembler for the ARM processor and assembled manually. When an infected file is executed, the virus displays a dialog box with the following two-line message asking for permission to infect:

```
WinCE4.Dust by Ratter/29A
Dear User, am I allowed to spread?
```

If granted permission, Duts attempts to infect all `.exe` files in the current directory. It only infects large files (larger than 4096 bytes) that are still uninfected. As an infection marker, the virus writes the string `atar` in the Version field of the `.exe` file header.

The virus infects a file by appending itself to the file and making the last section of the file readable and executable. The entry point of the `.exe` file is set to the beginning of the virus code. Duts contains two messages that are not displayed:

```
This is proof of concept code. Also, i wanted to make avers happy.
The situation when Pocket PC antiviruses detect only EICAR file had to end ...
```

The second message refers to the science-fiction novel *Permutation City* by Greg Egan, where the following sentence appears `This code arose from the dust of Permutation City`.

◇ **Exercise B.2:** What conclusion can be derived from this timeline?

> On a long enough timeline, the survival rate for everything drops to zero.
>
> —Chuck Palahniuk, *Fight Club* (1996)

Concluding Remarks

This chapter starts with a number of tips for increased security. It continues with a summary of malware and its most important features. The chapter ends with a few final conclusions.

This book has tried to instill in the reader both an awareness of and respect for computer security. Many computer users are aware of security problems simply because of what happened to them, but relatively few have a real respect for security and even fewer give this topic the time and effort it deserves. We therefore start with a short reminder, in the form of a list of security tips. These tips can be found elsewhere in the book but have been collected here as a parting gift from the author to those readers who have got so far in the book.

- Use strong passwords. Section 8.3 discusses passwords, their applications, and their weaknesses. It describes the features that a strong password should have and it presents examples of weak passwords. Also, passwords should be memorized, not written down, and they should be replaced often.

- Backup all your data regularly. Backups are discussed on page 143 as well as in other places in the book. This author would like to take this opportunity to stress again the importance of regular and full backups. All files should be backed up, including personal data, application programs, utilities, and operating system files. Those who have sensitive data should keep several generations of backup files, in case corrupted or infected files are discovered in the near future. One last word. If at all possible, check your backups. This is especially true for a large organization (government or commercial) that has sensitive or critical data whose loss may affect many users, customers, or citizens. Checking a backup is time consuming and requires extra equipment. The ideal setup for checking a backup is to have another computer, identical to the one whose files are backed up, and to actually run that computer on the backed-up data, executing programs and looking at data. This does not fully guarantee a clean backup, but will catch most corrupt files and data that had been diddled with.

- Obtain anti-virus software, update it, and use it regularly. Anti-virus software is

mentioned in Section 6.3 and in many other places in the book. While not 100% effective, this software is still the easiest and most cost-effective way to check for, discover, and delete viruses and other types of malware. However, as this book says in several places, it is important to have the latest versions of both the program itself and the virus update, and to run this software regularly (or at least every time a new virus update is released, which is typically 2–3 times a month). If the computer has a removable drive, it is important to set the anti-virus software to automatically check every volume inserted into the drive.

- Install a firewall and always use it, updating its rules as necessary. Even a simple firewall, just a small piece of software, considerably increases your chances of survival in the Internet jungle. Firewalls are discussed in Section 7.6 and are useful even to those who have a modem and have to dial a number to connect to the Internet.

- Be suspicious of email attachments. Section 2.4 lists tests that an attachment should pass before a careful user will consider opening it. Email attachments are regularly exploited as carriers of viruses and worms, and no one can count (or even estimate) the number of innocent computer users who became victims of malware by the simple act of clicking on an email attachment (often a love letter purportedly from a known and trusted friend).

  (A related principle is to close the preview pane of all email programs. Such a pane permits the user to read a message before it is opened, but can be abused by hackers.)

- Download all security patches available from software makers. When a security vulnerability is discovered in a widely-used program or operating system routine, its maker often issues a patch to correct the flaw in the software. It is important to use such patches because the existence of a patch doesn't deter hackers from trying to exploit a vulnerability. They know that many users don't install security patches (because of ignorance, laziness, apathy, or sheer plain stupidity) and they exploit this fact to achieve their aims.

- Finally, try to get in the habit of disconnecting your computer from the Internet as much as possible. Whenever you don't need your computer for communications, physically unplug it from your telephone, your modem, or your router. Many owners of personal computers run their computers continuously, and this may also contribute to security breaches. It is a good idea to either put the computer to sleep (a mode offered by all modern operating systems) or to turn it off completely when not in use (but see Section 4.2 about operating system maintenance done automatically late at night). Many computer users believe that a hard drive lasts longer if allowed to spin continuously, but consider the following: Magnetic disk prices (as well as prices of CD and DVD drives) are coming down all the time, while security risks, attacks, and threats are on the rise. With this in mind, a computer user should answer the following question: Is it better to have a long-lasting disk drive or to have complete backups and turn off the computer as much as reasonably possible?

> User, n. The word computer professionals use when they mean "idiot."
>
> —Dave Barry

Malware: Summary

The discussion of malware in this book can be summarized by looking at the many differences between rogue software and other types of software. There are three main areas where malware exhibits significant differences of behavior.

Generality. An attack (on a computer or a network) that does not involve a virus must be based on a weakness or a flaw in the object attacked. The attacker discovers that a certain security mechanism does not perform the right checks in certain cases, so the attacker creates such a case and thereby gains access to or control of the object. Such an attack generally allows the hacker limited access, so the damage must depend on the amount of access the attacker has.

A virus, however, spreads without exploiting any flaws or bugs in the protection mechanism of the computer, network, or operating system. The virus spreads when users share programs or other resources, and because the virus is a program it can cause any type of damage. We can think of a virus as a team of software installers that distribute a piece of software quickly and automatically. The software being distributed can be benign or malevolent, but the distribution mechanism is the same.

Range. The range of effect of a malicious program is much greater than that of other software. When an attacker breaks into a computer, he can read and delete all the files on that computer. The attacker may steal passwords and use them to break into other computers, but this has to be done manually, computer by computer. When an attacker breaks into a computer and installs a virus, the virus can infect many files on the computer and may propagate itself into other computers either by email, by files sent on the Internet, or by files written on disks and distributed to other computer owners. This feature implies that the range of effect of a virus is far greater than that of other (conventional) software. This feature also applies to benign viruses, which is why such viruses can be very useful.

Persistence. A program is easily deleted, but a virus may be difficult and time consuming to locate, delete, and completely eradicate. When a virus code starts executing, it may check the date, and set itself a target date of, say, three months in the future for releasing its payload. When the time comes to do its damage, the infected program (or programs) may have been backed up, perhaps several times, on several backup devices, for three months. The result is that many copies of the virus may have found their way into all the backups of the computer owner, with the unfortunate consequence that just deleting the virus from the computer is not enough. Keeping up-to-date backups is important, but the backup device itself should be checked for viruses before it is used to restore any infected files.

Perhaps the first thing that comes to mind when an infected program is discovered is to delete it and replace it with a clean copy. Often, this step is performed after the virus has already spread throughout the computer and has infected other programs, with the result that the clean copy is going to be infected soon.

A virus is also persistent because it may find its way to removable storage, such as floppy disks and zip cartridges. In a university environment, it is common to have a computer lab where the computers are connected in a local-area network. It is also common for users to keep files on removable disks and insert the disks into different

computers to run programs. Once a virus appears in a file in a computer in such a lab, it will propagate into other files, then into other computers on the network, and then into removable disks. A security person may spend much time cleaning all the computers in the lab, only to have the virus appear again as soon as a user inserts a removable disk into a drive.

A policy that should be adopted in such an environment is to keep the important files (applications, utilities, and operating system) in a locked (read-only) part of the hard drive of each computer, and let the users store their data files temporarily in the unlocked part of the disk. When those files get infected, the unlocked part is simply erased. Another solution is to have computers with no hard drive, and to serve files from a central server that is well-protected by a firewall and by experienced security personnel.

An Internet search unearths real examples of viruses that persisted in an environment for months and kept coming back after each thorough cleaning of the computers involved. An example is the scores virus (Section 5.6).

Over the years, computer users have noticed that viruses written for old versions of an operating systems linger on in the computer even after several newer versions of the operating system have been installed. It is common for computer makers to base a new computer on older models, in order to maintain upward compatibility. This is a useful feature that allows old software, which represents a substantial investment, to run on a new computer, but it also means that viruses written for the old computer may find their way into the new computer and continue their destructive mission for years.

◇ **Exercise Conc.1:** What are well-known examples of computer families with upward compatibility?

Persistence is perhaps the most important feature of benign viruses. Once such a virus enters the environment of a computer or a network, it does its job for years without any supervision or maintenance.

Final Conclusions

Computer security is a vast area that affects the performance of businesses, the quality of services provided by governments, and the daily lives of many on Earth (and perhaps elsewhere in the universe). Security is steadily becoming both more complex and a bigger threat. Security is getting more complex because more security holes are being discovered and because operating systems are becoming more complicated. Security is becoming a bigger headache because of the prevalence of computer networks. The root of all computer security problems is the inability of computers to distinguish good from bad. On a slightly lower level, security threats exist because of the existence of networks and the complexity of modern software. Computer security threats normally arrive at a computer from the outside through a network and they (the problems) thrive on software vulnerabilities.

One natural conclusion from the previous paragraph is that an organization that decides to connect a local-area network of computers to the Internet should stop for a moment and consider the security aspects of this step and how best to handle them.

Similarly, an individual wanting to connect their personal computer to the Internet should first study the security ramifications of this step and be prepared to deal with threats when they arise. Another conclusion is that software users have to balance the advantage offered by new, complex software with the increased security threat that such software poses.

Malware, spam, email dangers, and spyware are bad, but not all bad, as the following quotation shows (by an anonymous writer identified only as `Floydian_99@yahoo.com` and referring to what that writer did while his computer had to be cleaned from viruses or spyware, I forget which).

> Things are not so sad after all, because this break gave me the time to come up with this document. I hope this will stand as a must read for network administrators and security experts out there. As new technologies and new viruses will emerge, some of the information may soon be obsolete, but I think...

Another important conclusion from this book is that all security is compromise. It is possible to be very secure, but this can be achieved only at the cost of making the computer less convenient to use. Every security measure, technique, and device results in a slower, less responsive, and clumsier computer. Once this is realized, each computer user has to decide how much security they need. Performing a virus check and file backup every day increases security, but is time consuming. Looking at each email attachment, examining it, thinking about it, and applying tests to it likewise increases security, at the cost of time spent (and higher blood pressure of the user). Installing an activity monitor that detects suspicious or unusual activities also beefs up security, but decreases the user's "quality of life" at the keyboard because of the need to respond to the monitor's discoveries and questions and to constantly make decisions. Even the process of reading this book boosts security by giving the reader confidence, but has the downside of taking time. (Unless you listen to it while you sleep. Ask the publisher to come up with an audio version of the book. Just kidding.)

◇ **Exercise Conc.2:** The previous paragraph says "all security is compromise," implying that compromise is a general attribute that underlies any type of security. Show an example of this compromise in a non-computer situation.

The goal of this book is to familiarize you, the reader, with the reasons for security threats and with the best security procedures and practices currently available. You should worry about the security of your computer, and this book is trying to teach you to worry correctly. Whether or not this goal is ever reached in your environment is up to the individual reader (but you can still complain to this author when something goes wrong, just to get it off your chest).

Final conclusion: Practice safe computing.

A self test. The following questions will help you decide whether you have read this book carefully.

1. What is the human factor of computer security? (See page xiv.)
2. What does the word "system" mean? (See page xvi.)
3. Why does the problem of computer security exist? (See page 4.)

4. What is the worst thing that can happen to computer security? (See page 8.)
5. How should a computer user/owner start each day? (See page 9.)
6. How should we look at security? (See page 10.)
7. What is the best line of defense against all types of computer security threats? (See page 13.)
8. Does the sentence "people are nosy and machines are noisy" sound familiar? (page 15.)
9. What are the various types of malware?
10. How does a computer virus make sure it gets executed in an infected computer?
11. How can an international organization help in the war against malware? (See Section 3.3.)
12. How safe am I if I have a complete backup of all my files? (See Section 6.4.)
13. Why are virus hoaxes bad? (See Section 6.5.)
14. How do spammers obtain so many valid email addresses? (See Section 7.4.)
15. Can spammers be defeated by legal means? (See case study on page 178.)
16. Does a firewall consist of hardware or software? (See Section 7.6.)
17. What is the most secure biometric authentication technique? (See Section 8.2.)
18. Why is a shredder such a useful tool in the computer security war? (See Section 10.2.)
19. What are the main security concerns of parents? (See Section 11.3.)
20. Why do we trust certain Web sites and mistrust others? (See Section 11.4.)
21. What does the security of an encrypted message depend on? (See Section 12.2.)

It's a rash man who reaches a conclusion before he gets to it.
—Jacob Levin

Answers to Exercises

> A bird does not sing because he has an answer, he sings because he has a song.
>
> —Chinese Proverb

Pre.1: Auditing the auditors is fairly easy because the auditors don't write any software and are not supposed to modify existing software. Thus, auditing the auditors is done by making sure they haven't made any changes. This is a straightforward task that can be done by one person who is trusted by the owners/directors of the bank.

Pre.2: One approach to this problem is to sabotage the computer from time to time by creating a short circuit. When the technicians arrive in the computer room, you can cut short your shift and go home. The author isn't recommending such a solution, but similar stories (some perhaps true) have been circulating in the computer security community for many years.

Intro.1: The car industry. A modern car has several computers that control its operations and sense failures. As a result, many millions of small, specialized computers are purchased and installed by car manufacturers every year. Fortunately, there haven't been yet any security problems with those special, embedded computers, but they have become themselves a major source of car trouble.

Intro.2: The question is meaningless. A computer is a machine and as such is neither trustworthy nor untrustworthy. These terms are attributes of humans, which implies that trusting a computer really means trusting those who designed and built it.

> The question of whether computers can think is just like the question of whether submarines can swim.
>
> —Edsger W. Dijkstra

Intro.3: A hacker who has physical access to your computer can replace your keyboard with a rigged one that has a radio transmitter. The hacker would then receive and record all your keystrokes even if you check for spyware and remove all of it.

Intro.4: Your accent can tell much about your origin, as the following quotation illustrates.

> Simply phonetics. The science of speech. That's my profession; also my hobby. Happy is the man who can make a living by his hobby! You can spot an Irishman or a Yorkshireman by his brogue. I can place any man within six miles. I can place him within two miles in London. Sometimes within two streets.
>
> —George Bernard Shaw, *Pygmalion* (1916).

Intro.5: One example is the following: Security holes and weaknesses lurk everywhere and no amount of user testing and expert thinking can find them all. They are discovered slowly, one by one, but every new piece of software written and every new piece of hardware built may contain a new security flaw.

1.1: Yes, because of two reasons. (1) Most passwords are short enough such that 20% of the password is just one character, which makes it easy to use a brute-force approach to guess the missing character. (2) A computer user may enter the same password several times a day, making it trivial for a spy to complete a missing character.

1.2: Two things. A hard disk may crash and files may be left open and become inaccessible as a result.

1.3: Because basements are prone to flooding.

1.4: The first step is to consult a `whois` data base such as `www.arin.net/whois` to locate the owner of the IP number in question. This is either an ISP or a large organization that has been assigned a block of IP numbers. The second step is to convince the owner to identify the user located at the IP number.

1.5: Given an unknown data item A, if we repeatedly add to it random numbers that are distributed normally with mean m, we end up with random numbers that are distributed normally with mean $m + A$, thereby allowing an accurate estimation of the unknown data.

2.1: A personal computer may be used by several users (such as the members of a family). At any given time, only one user can use the keyboard and display, but in principle a user can log in, start a program, and leave, only for another user to log in and start another task. Thus, the operating system of such a computer should allow each user access only to their files and should be able to allocate time slices to several programs. However, an operating system on a personal computer can restrict the use

of the computer to one user at a time (a user has to log off before another user can log on) which means that only the programs of one user can reside simultaneously in memory. Even in such a case, the operating system has to protect each program from all the other ones. Thus, the answer is yes, a personal computer can be considered a multiuser computer, with the single exception that only one user sits at the keyboard at any time.

2.2: Here is one in Java (it should be typed on a single line).

```
class s{static public void main(String[]x){String a="class s{static public
void main(String[]x){String a=;System.out.print(a.substring(0,52)+(char)34
+a+(char)34+a.substring(52));}}";System.out.print(a.substring(0,52)+(char)34
+a+(char)34+a.substring(52));}}
```

2.3: Most computer peripherals have moving parts and can be damaged by forcing them to repeat certain operations many times. Here are some examples.

- A CD (compact disk) is normally read-only, but there are recordable CDs (CD-R) that can be recorded once by the computer, and even recordable rewritable CDs (CD-RW), that can be recorded, erased, and reused by the computer many times. Malicious software can erase such a CD every time it is inserted into the CD drive. Even worse, it can erase the CD many times in a short period of time, thereby shortening its life.

- The popular flash memories can be damaged in the same way.

- An uninterruptible power supply (UPS) is the next example. Such a device uses a line-voltage battery to support the computer for a short time in case of a power failure. Recent UPS devices may be connected to the computer with a cable (normally USB) and can launch a utility that closes all the active applications and open files, and turns the computer off when the battery runs low. Malicious software can corrupt this utility such that it opens many files and continually spins the hard drive. When battery power runs out, there is a good chance that the read/write head of the hard drive will crash, thereby physically damaging the drive.

- Old dot-matrix printers had many small moving parts and it was possible to damage the printing head and the platen in such a printer by printing a dot pattern, backspacing the printing head, and repeating this many times.

- A CRT has a screen coated with a phosphor compound. When a beam of electrons hits the screen, it is stopped and its kinetic energy is converted to visible light. Screen savers are programs that move the beam continually when the computer is not in use, or simply dim the screen, to make sure that no point on the screen will be exposed to the beam for a long time. Malicious software can corrupt the screen saver so it concentrates the beam at one point on the screen long enough to burn the phosphor coating at the point, and then move the beam to another point, to repeat the damage.

- Modern personal computers have several cooling fans in them. The fans are turned on and off by the operating system depending on the temperature at various points

inside the computer. Rogue software can interfere with this operation in an obvious way and can seriously damage the computer as a result of high temperature.

- A computer virus in a laptop can spin the disk continuously and drain the battery very quickly. This does not damage the computer, but is annoying. A similar virus in a cell telephone can achieve the same result even in the absence of a disk.

2.4: When a disk is infected, the virus saves the original boot sector to a different location L on the disk. If an infected disk is reinfected, the virus would save the *modified* boot sector to L. Thus, any virus detective trying to read the boot sector would read the infected version, whether it came from its original position or from L. This would make it easy to identify the virus.

2.5: An extra track may be used for copy protection. Only utilities that know about the track can fully copy the disk.

2.6: The Pareto principle (also known as the 80–20 rule) is named after Vilfredo Pareto, an Italian economist, and was popularized by Joseph M. Juran. It claims that 80% of the results of an operation often stem from 20% of the causes of the operation. This is a useful idea that can be employed as a rule of thumb in many situations, but can also be misused. Examples of this rule are: (1) 80% of the real properties in a certain region may be owned by 20% of the population. (2) 80% of the sales of a company may be due to 20% of its customers. (3) 80% of the execution time of a computer program is caused by 20% of its instructions. However, the claim "80% of the work is done by 20% of the workers" is, in general, wrong.

2.7: The virus selects a few bytes from the middle of the program, and replaces them with a jump instruction to the virus. When execution of the program gets to the jump, it jumps to the virus, which then executes, restores the selected bytes, and resumes the program. This method makes it difficult to detect the virus, but calls for an experienced virus writer, because modern computers have variable-size instructions (an instruction may occupy one byte or several bytes) and the virus has to place the jump between two original instructions, not inside an instruction.

2.8: If the virus came from the boot sector of a removable disk, then the computer has a drive for removable disks and there is a chance that more removable disks will be inserted in the future. If such a virus tries to infect only executable files, it will miss all the future removable disks. On the other hand, if the virus is residing in an executable file (it is a file infector), the computer may not even have a floppy drive or a zip drive, so removable disks will never be inserted and the virus will never propagate.

2.9: The date is different, but the word processor (or other software) has a command, such as "`\date`," to obtain the date from the operating system, which is why the header is always the same.

2.10: A virus may do nothing because of a bug in its code. Some viruses are written by researchers as a proof of concept, to study a certain aspect of virus propagation or infection, and such a virus may also be benign.

2.11: Changing one bit in a text file modifies one character of text. Often, this may not constitute significant damage, especially since the text in question may have mistypes to begin with. A corrupted character in a poem may never be discovered, even by its author, but legal or medical texts may be sensitive to even small corruptions. Modifying one bit in an image changes the color of one pixel. If the color changes significantly, the modification may be noticeable. In medical (X ray) images and images taken by spy satellites, every pixel may be important, and a corrupt pixel may lead to wrong diagnosis (in the former case) or wrong military decision (in the latter). It's difficult to think of a case where one bad bit in a video file would be noticeable, but a single bad bit in an audio file may be noticed if it changes a short interval of silence to a loud sound. Changing one bit in an executable file corrupts either an instruction or a data item. In either case the program will get corrupted, but may still do its job most of the time. We know that a typical program spends most of its time in small regions (loops) of instructions, while most of its instructions are rarely executed. If one instruction in an error-handling procedure is damaged, the program will still run correctly until the error actually occurs.

2.12: The virus may go into action when the user closes a document in a word processor. Before the virus closes the document, it may make random changes in the text. A sophisticated data diddling virus may simply check every text file found in the computer, scan it for predetermined keywords or phrases, and change each. Thus, each occurrence of "vice president" may be changed to "president" and each "buy" changed to "sell." Another nasty idea is to search for spreadsheet files and change them along the same lines. A file may be changed each time it is saved, or only when the user closes it, or when the virus finds it on the disk. The virus may modify data items or formulas; it may follow guidelines (such as change every "+" to "−" or swap every data item with the one to its left) or may do its damage at random. The point is that it's easy to come up with ideas for inflicting subtle damage, but it is difficult, perhaps even impossible, to correct such damage once it is discovered.

2.13: A vice-president trying to get rid of the president quickly and fill his place. One partner trying to drive the other partner crazy and buy his share of the business awfully cheap. A student trying to have a fellow student fail a class project.

2.14: If a low-clearance user can see the names of high-level files, then S can temporarily modify the name of such a file to signal a 0 or a 1 to R. If S can control the existence of a shared resource, then it can signal a 0 or a 1 by the presence or absence of the resource. Similarly, S can send bits to R by deleting and creating a file each time it receives a synchronization signal from R.

2.15: Because in general it is easier to do evil than to do good, as can be seen by many real-life examples. It takes years to build a large building, but only seconds to bring it down in an earthquake. Similarly, raising a child requires years of effort, but killing someone is much easier and quicker.

2.16: Yes, if the executable file is small to begin with or if it does not compress very well (which happens if it is random or close to random).

2.17: Yes, but I promise to do my best to educate myself and follow the examples, tips, and advice given here and in many other books, articles, and Web sites to try to keep my computer, backups, and network clean.

2.18: (1) Execute a `clear` instruction. Such an instruction clears its operand, which may be a register or a memory location. (2) Subtract register 4 from itself. (3) Prepare the constant zero in location `cons` and execute a `mov` instruction to move that location to register 4. (4) Multiply register 4 by zero. (5) Shift the register n positions to the left or to the right, where n is the register size. (6) Perform an exclusive OR (XOR) of the register with itself.

2.19: On the author's Macintosh, the activity monitor indicates the following: Alias menu (displays a menu of files at the top of the screen, for easy launching), TypeIt4Me (a simple macro processor that types a character string when the user types its name), fax assistant (looking for incoming faxes), Iomega driver (looking for a zip disk inserted into the drive), and Little snitch (a firewall program).

2.20: When a virus senses that new, unfamiliar software has been installed and is scanning disk directories or memory, the virus may react by erasing several important operating system routines from memory (and also deleting them from the disk). This leads to a crash, where the computer behaves erratically or seems frozen. Such a stealth technique is extreme and immediately raises suspicion of a virus, but it may delay the detection and extermination of the virus, or at least may annoy the computer user a while longer.

2.21: Restarting (or rebooting) a computer is done by an interrupt. The user presses a restart button (or a key combination) that generates a special interrupt, and the handling routing for that interrupt closes all the open files and restarts the computer. (On a PC, the key combination CTRL-ALT-DEL is used for this purpose.) This is why a computer can be restarted at any time, even in the middle of a program, and even if the program is stuck in an infinite loop. A virus that infects the reboot interrupt-handling routine can therefore survive a restart. The virus copies itself as a temporary file on the disk, it modifies the operating system routine that boots the computer, and then executes the normal routine for closing down the computer. When the computer restarts, it executes the modified booting routine. The routine boots the computer normally, and then reads the virus from the temporary file, stores it in memory, and deletes the file. As a precaution, the virus may modify the booting routine such that its last step is to remove the modification, leaving nothing suspicious behind.

3.1: Try to obtain it from the maker of the vulnerable software. A large software maker that sells, for example, a web server (where the hacker has discovered a security hole) will have a list of customers who purchased the software. Such a list will not have the IP numbers, but they can be obtained automatically from the URLs of the customers. The list will not be perfect, but it doesn't have to be.

3.2: One such example is a dictionary server. The client sends a word, and the server responds by sending back the definition of the word, or its synonyms. Another example is a street-address server, such as [mapquest 04] or [maporama 04]. The client sends the latitude and longitude coordinates of a point on Earth, and the server sends back the street address, if any.

3.3: It is true that any experiments with rogue software, not just worms, should produce best results if carried out "in the field," rather than in a laboratory. However, the public would have to be notified, and there will be tough resistance to such an experiment, because people are scared of anything they don't understand. Also, the good worm may have a bug in it, that will turn it into a bad worm. Even worse, a hacker may find a way to release a private, bad worm, similar to the good one during the experiment and that worm would spread with the good one, finding no resistance.

3.4: When a worm generates children and sends them out, each child should get a list of IP addresses as before, but also including the addresses of its older siblings. Thus, if a worm generates children A, B, and C, then B will get a list that has, among other addresses, the address of A, and C will get a list with both A's and B's addresses.

3.5: We know that it's virtually impossible to completely test and debug a worm (or any other type of rogue software) in the laboratory. Thus, once a worm has been sent into the Internet, its creator may discover a bug in the code. Another reason may be a worm's author who has just finished reading this section and is eager to employ the techniques found here to improve an existing worm.

4.1: A word processor or an editor runs with a user's own file access privileges, because it must have full access to all the user's files. A Trojan horse in a word processor can therefore read, write, or delete any file opened by the word processor. The horse could, for example, examine the content of the file, and upon finding a keyword (such as `key` or `bomb`), send the entire text to its creator, create a copy that's publicly accessible, or change the access permission of the file to make it generally accessible.

4.2: Many of the spyware applications discussed in Chapter 9 are Trojans.

4.3: There aren't many. Perhaps the most important ones are the `mail` routine, the `passwd` routine (to let users change their passwords), the `ps` command (examines the status of all processes in the computer), routine `lquota` (to enforce disk quotas), and the `df` command (which indicates the amount of free disk space).

4.4: The code of line 3 identifies the fact that the compiler is compiling itself, but this code is weak, because even minimal changes to the source line "`compile(s) {`" will defeat the test of line 3.

4.5: The original virus can check to see whether a file with a certain name exists. Once the hacker decides that the original virus has done its job, he creates such a file. Whenever any copy of the virus notices the existence of this file, it deletes itself from its host. A variation is to have the original virus itself create this flag file once it has infected the compiler and thus accomplished its mission (see the discussion of antibodies on Page 73).

6.1: A dentist's office, as discussed on page 144.

6.2: Yes, to some extent, because many new viruses are created from virus kits and are therefore modified versions of existing viruses, so their codes are similar. Old versions of Norton anti-virus software for the Macintosh, made by Symantec, promised to do just that.

6.3: An image file tends to be big and is almost always kept in compressed form. Even a small image may consist of a million pixels, each occupying three bytes. Current digital cameras are already in the six megapixel range, and create image files that are at least 18 Mbyte long in uncompressed form. Video files are much bigger. A document with tax information may be needed only once a year. X-ray images in a hospital's archive may remain untouched for years. Pictures taken by astronomical telescopes are stored in archives and may be used years later to compare a newly-discovered astronomical event with the same patch of sky in the past.

6.4: Because the activity monitor is called by the break routine and this call places another return address at the top of the stack. Thus, when the activity monitor is executing, the address at the top of the stack is the address the activity monitor will use to return to the break routine. The address below it is the address the break routine will use to return to F.

6.5: Many applications are installed by an installer that decompresses the files needed by the application and writes them in the appropriate places on a disk. When such an installer finds files left from an older version of the application, it may get confused. If the access permission of a folder has changed and the installer can no longer write to it, it may skip part of the installation or terminate abnormally. If power to the computer is cut off during the installation, the installer may not be able to complete its task later.

6.6: A screen saver. See Section 9.3.

6.7: When a program terminates normally, it returns control to the operating system by creating an artificial interrupt called a break or a supervisor call. When the hardware senses this interrupt, it invokes an interrupt service routine (part of the operating

system) that either examines user commands that are pending or decides what program will be next to execute. If the virus writer is familiar with the details of the operating system, the virus may modify this routine such that it first infects whatever program has just finished, then executes normally. Notice that in order to infect an operating system routine, the virus has to obtain high privilege, but such viruses have been detected in the past.

6.8: The voting circuit cannot decide which result is correct, if any. In such a case, the circuit can only detect an error and cannot correct it. An improvement is to have five copies, or even a larger (odd) number of copies.

7.1: Because the association between a URL and its IP may change at any time, especially if the site in question is hosted by an ISP that provides dynamic IPs.

7.2: Cheap (but possibly bad) prescription drugs, drugs that enlarge or enhance body parts, herbal remedies, weight loss drugs, get-rich-quick schemes, and financial services such as mortgage offers or schemes for reducing debts. Qualifications, such as university degrees or professional titles. On-line gambling. Cut-price or pirated software.

7.3: Just do it.

7.4: At the time of this writing they are `freebieland.net`, `coolfreebielinks.com`, and `freebielist.com`.

7.5: The term zombie is used in UNIX to indicate a child program that was started by a parent program but was later abandoned by it. This is not the same as a zombie computer or a zombie server.

8.1: If you know a person, you can ask him an array of personal questions. If you are satisfied with the answers, you authenticate the person. If you don't know a person, you can receive the answers beforehand, and conduct the authentication process by computer, but this method is still experimental and should not be trusted.

8.2: A natural eye can be distinguished from a glass eye by shining light of varying intensities on the eye and making sure that its pupil dilates normally.

8.3: Another variation is to prepare a large number of different permutations, then compute numbers a, b, and so on from the password, and finally perform permutation a on the password, then apply permutation b on the result, and so on.

8.4: Typical default passwords are the following: help, test, tester, system, system, manager, sysman, sysop, engineer, ops, operations, central, demo, demonstration, aid, display, call, terminal, external, remote, check, net, network, phone, and fred.

8.5: Yes, it is secure, because there are so many possible permutations. However, it may be easier to memorize a random password than to memorize a specific permutation of 16 characters, again because there are so many permutations.

8.6: It is likely `';lkjh` or `lkjhgf` (look at your qwerty keyboard).

8.7: Products are continually becoming more reliable, easier to use even with a personal computer, and less expensive. Use a search engine and search, for example, under "fingerprint identification."

8.8: Ask a friend to let you use their account. Try to guess a password by using the birthday of its owner, the number of his children, or their birthdays.

8.9: The name of the person, as in Larry's case.

8.10: In the case in question, the hacker identified a computer that ran an early version of `PC AnyWhere`. This software [remotelyanywhere 04] makes it easy to remotely control a PC and the early version had a security flaw that made it possible to login to a remote PC while bypassing the password protection (a classic example of a security compromise). Once gaining control of the computer, the hacker identified its IP address, then used `telnet` software to try nearby IP numbers to break into other computers on campus.

9.1: Are there any complaints on the Internet, especially on popular technical message boards, about the program using deceptive advertising or spam?

9.2: No. If this poor, young, and inexperienced author could come up with such a frightening scenario, imagine what a group of well-determined, well-funded, and well-trained terrorists could come up with when they really put their minds to it. The only remedy to this scenario is for a government agency to check the background of every affiliate network and to scan for a Trojan horse every computer program that seems too good to be true; not very practical.

9.3: A search in early 2005 discloses that James Carter, of Lilburn, Georgia, 30047, an unemployed cardinal health worker, has made two contributions totaling $500 to the campaign of Howard Dean, a 2004 presidential candidate and former governor of the state of Vermont.

9.4: This is easy. A search for "spyware audit" has returned about 260,000 results.

9.5: This is easy. A search for "spyware removal" returns several million results, among them [Spybot 04], makers of *SpySweeper*; [snapfiles 04], that advertises *Spyware Doctor*; [lavasoft 04] with its *Ad-Aware*; and [SearchAndDestroy 04] that offers free spyware removal.

10.1: This author cannot think of any.

10.2: (1) Open a text file, copy the individual characters of the password one by one, and paste each character where you need to type the password. (2) Start by typing a string of `a`s, then type the characters of the password in random order in between the `a`s, and finally delete the `a`s with the delete key. As an example, consider password `ChHaakon`. It can be typed surreptitiously in the following steps

`aaaaaaaa` → `aaaaakaaa` → `aaaHaakaaa` → `aaaHaakaoaa` →
`aCaaHaakaoaa` → `aCaaHaakaoana` → `aCahaHaakaoana` → `ChHaakon`.

(3) Use a virtual keyboard. This is a program that displays a keyboard on the screen. The user clicks on keys, and the corresponding characters are displayed at the cursor's location. An example of a virtual keyboard is [corallosoftware 05]. These methods are safe but tedious, which illustrates the tradeoff between security and ease of use.

10.3: Here are some examples: (1) A bank statement with cancelled checks. A sophisticated thief can wash off any traces of ink from a check, then use it in an obvious way. (2) A box full of newly-printed checks. (3) A preapproved credit card offer. It is always a good idea to opt out of such offers and in the United States this is possible by calling 888-5-OPT-OUT. (4) New credit cards issued for old, expired ones. (5) Letters (with checks, official forms, or money orders) you leave in your mailbox for the mailman to pick up.

10.4: Marketers and spammers send unwanted advertisements and spam targeted by IP numbers. Hackers and snoops track a victim's Internet surfing habits by his IP address, which increases one's chances of becoming a victim of identity theft.

10.5: Profiling and targeted advertising are popular applications of cookies. You surf to a magazine's site and decide to read an article on weight loss. The site sends your computer a cookie that identifies you as one who is interested in weight loss. On every subsequent visit to the magazine's site, the cookie is read by the site, which then displays ads for weight loss.

The biggest online advertising company is DoubleClick. Relatively few have heard of this company and even fewer have visited their Web site. However, chances are that you have a cookie in your browser from DoubleClick even if you have never visited that site [cookiecentral 04]. This is a third-party cookie, sent by a site that you have visited. When you visit a commercial Web site `X` that employs DoubleClick as its online advertising company, `X` retrieves any DoubleClick cookies that you may have and sends them to DoubleClick, which then sends `X` ads based on those cookies for you to watch. Site `X` then sends you another DoubleClick cookie identifying you as a visitor of `X`.

10.6: By spreading false rumors about a new, revolutionary product about to be released by Microsoft and including a link to his site for anyone to click on instead of typing.

10.7: You cannot. A search at [Network solutions 04] indicates that this domain has already been registered (by Alon Swartz) and so have virtually all the domain names that are typographically similar to `microsoft.com`.

11.1: A news agency may want to customize news it carries in its Web site by time zones.

11.2: A redundancy. Similar to phrases such as absolutely necessary, advance warning, boiling hot, hot water heater, my personal opinion, and newborn baby.

11.3: Yes, because a child, especially a pre-teen, may not fully grasp the risk of opening an email attachment or may easily forget any warnings they received about this danger.

11.4: Here is an example of such a set.

- Search the Internet for Web sites that carry news about new malware. Browse 2–3 such sites every morning. The few minutes that this takes are time well spent. Search the Internet for security news and white papers on security.
- Obtain a firewall, use it, and update its rules as needed.
- Obtain anti-virus software, update it and run it regularly.
- Shut down your computer when you are not using it.
- Enable other operating system services only when necessary and only on a temporary basis.
- Use robust passwords that include letters, digits, and other characters in a hard-to-guess string. Remember! Sophisticated dictionary attacks are being carried out all the time.
- Keep up to date with security patches for (1) your operating system, (2) your Web browser, and (3) other software, especially any servers or communications software.
- Keep a sharp eye out for domain name trickery.
- Never download anything from any source you don't know and trust, and be sure it really is the source you think it is. As this book explains elsewhere, it's easy to create a convincing fake version of any Web site.
- Delete, without reading or opening, email attachments from anyone you don't know and trust. Be wary of attachments even from known, trusted persons, whose computers may have been compromised.

12.1: The number of 64-bit keys is $2^{64} = 18{,}446{,}744{,}073{,}709{,}551{,}620$ or approximately 1.8×10^{19}. The following examples illustrate the magnitude of this key space.

1. 2^{64} seconds equal 584,942,417,355 years.
2. The unit of electrical current is the Ampere. One Ampere is defined as 6.24×10^{18} electrons per second. Even this huge number is smaller than 2^{64}.

3. Even light, traveling (in vacuum) at 299,792,458 m/s, takes 61,531,714,963 seconds (about 1,951 years) to cover 2^{64} meters. This distance is therefore about 1951 light years.

4. In a fast, 5 GHz computer, the clock ticks five billion times per second. In one year, the clock ticks $5 \cdot 10^9 \cdot (3 \cdot 10^7) = 1.5 \cdot 10^{17}$ times.

5. The mass of the sun is roughly $2 \cdot 10^{31}$ kg and the mass of a single proton is approximately $1.67 \cdot 10^{-27}$ kg. There are therefore approximately 10^{58} protons in the sun. This number is about 2^{193}, so searching a keyspace of 193 bits is equivalent to trying to find a single proton in the sun (ignoring the fact that all protons are identical and that the sun is hot). The proverbial "needle in a haystack" problem pales in comparison.

6. The term *femto*, derived from the Danish *femten*, meaning *fifteen*, stands for 10^{-15}. Thus, a femtometer is 10^{-15} m, and a cubic femtometer is 10^{-45} cubic meters, an incredibly small unit of volume. A light year is 10^{16} meters, so assuming that the universe is a sphere of radius 15 billion light years, its volume is $(4/3)\pi(15 \times 10^9 \times 10^{16})^3 = 1.41372 \times 10^{79}$ cubic meters or about 10^{124} cubic femtometers. This is roughly 2^{411}, so searching a keyspace of 411 bits is like trying to locate a particular cubic femtometer in the entire universe.

These examples illustrate the power of large numbers and should convince any rational person that breaking a code by searching the entire keyspace is an illusion. As for the claim that "there is a chance that the first key tried will be the right one," for a 64-bit keyspace this chance is 2^{-64}. To get a feeling for how small this number is, consider that light travels 1.6×10^{-11} meters (about the size of 10 atoms laid side by side) in 2^{-64} seconds.

12.2: Follow each letter in the key `polybiuscher` with its first successor that is still not included in the key. Thus, `p` should be followed by `q` and `o` should be followed by `p`, but because `p` is already included in the key (as are `q`, `r`, and `s`), the `o` is followed by `t`. This process produces first the 22-letter string `pqotlmyzbcikuvswhnefrx` which is then extended in the same way to become the 25-letter string `paqdogtlmyzbcikuvswhnefrx`.

12.3: Yes, as is easy to see by examining the following examples (notice the two occurrences of 22 in the ciphertext and how they produce different plaintexts):

Plaintext	+	66	05	66	11	61	Ciphertext	−	**22**	**61**	**88**	**22**	**27**
Key		66	66	22	11	66	Key		66	66	22	11	66
Ciphertext		**22**	**61**	**88**	**22**	**27**	Plaintext		66	05	66	11	61

12.4: The average word size in English is 4–5 letters. We therefore start by examining 4-letter words. There are 26 letters, so the number of combinations of four letters is $26^4 = 456{,}976$. A good English-language dictionary contains about 100,000 words. Assuming that half these words have 4 letters, the percentage of valid 4-letter words is $50000/26^4 \approx 0.11$. The percentage of 5-letter words is obtained similarly as $50000/26^5 \approx 0.004$. Random text may therefore have some short (2–4 letters) words, and very few 5–6 letter words, but longer words would be very rare.

12.5: When an encrypted message is sent by Alice to Bob, it can be intercepted by Eve and copied. When the key is later sent, Eve may intercept it and use it to decrypt the message.

12.6: Mixing salt and pepper is a one-way operation in practice (in principle, they can be separated). Heat flow from high to low temperature in a closed system is a one-way process in principle. Giving birth is one-way in principle, while squeezing glue out of a tube is one-way in practice.

12.7: This is a direct result of the properties of the modulo function. In step 3, Alice computes

$$\beta^a \bmod 13 = (5^b \bmod 13)^a \bmod 13 = 5^{b \cdot a} \bmod 13,$$

and Bob computes the identical expression

$$\alpha^b \bmod 13 = (5^a \bmod 13)^b \bmod 13 = 5^{a \cdot b} \bmod 13.$$

12.8: The final key is computed, in step 3, as $L^{a \cdot b} \bmod P$ (or, identically, as $L^{b \cdot a} \bmod P$), so it is an integer in the range $[0, P-1]$. Thus, there are only P possible values for the key, which is why P should be large. If we allow values L greater than P, then a user may accidentally select an L that is a multiple of P, which results in a key of 0, thereby providing an eavesdropper with useful information. If P is a prime and if $L < P$, then P is not a prime factor of L^x, so $L^x \bmod P$ cannot be zero.

A.1: No answer provided, but if you cannot easily read that (and I hope that you cannot), then you are not a good hacker.

B.1: Imagine an anti-virus program that does not yet recognize `Dark Avenger.1800`. The program works by opening executable files and checking them, with the result that they all become infected even though they are not executed.

B.2: Grim.

Conc.1: The IBM S/360 family was succeeded by the S/370, and 3080, 3090, and 43xx families, all upward compatible. The DEC PDP-11/20 was the first model (made in 1970) of the famous PDP-11 family of upward compatible mini and microcomputers, whose last descendant was the Micro PDP-11/94 (made in 1990).

Conc.2: Banks have to transfer large amounts of cash and they spend much effort and money on securing these transfers. They employ specially-designed armored trucks, trained security personnel, and special cameras and other security equipment. It would be easier and cheaper (and not much slower) to simply use a delivery service such as Federal Express to perform this task, but it would also be extremely nonsecure. The compromise in this case is extreme.

Compromise

Noun:
1. A middle way between two extremes.
2. Accommodation in which both sides make concessions.

Verb:
1. Make a compromise; arrive at a compromise.
2. Settle by concession.
3. Expose or make liable to danger, suspicion, or disrepute.

The only exercise some people get is jumping to conclusions, running down their friends, side-stepping responsibility, and pushing their luck!

—Anonymous

Glossary

Access control. Safeguards that prevent unauthorized access to a computer or a computing facility. An access control can be physical, such as lock or guard, or software-based, such as a password or a firewall.

Account harvesting. The process of collecting account names and passwords on a computer or a database.

Active content. Executable code (often in Java) embedded in a Web page. When the page is read, downloaded, and displayed by a Web browser, the embedded code is executed and may release a harmful payload.

ActiveX. A technology that extends the capabilities of a web browser (from Microsoft).

Activity monitor. Techniques that attempt to prevent malware infection by looking for suspicious or unusual activity in the computer.

AES. Advanced Encryption Standard, adopted by NIST as a replacement for the DES.

Anti-virus software. Software that searches for viruses and other malware. (See also Heuristic scanner.)

Applet. A small application. This term normally refers to Java applets.

ASCII code. (although Unicode is becoming a competitor). ASCII stands for American Standard Code for Information Interchange. It is a $(1 + 7)$-bit code, meaning 1 parity bit and 7 data bits per symbol. As a result, 128 symbols can be coded. They include the upper- and lowercase letters, the ten digits, some punctuation marks, and control characters. (See also Byte, Unicode.)

Asymmetric algorithm. A cryptographic algorithm where different keys are used for encryption and decryption. Most often a public-key algorithm. (See also Public-key algorithm.)

Attachment. Any file, data or executable, attached to an email message.

Attack. (1) An approach used by a codebreaker to decrypt encrypted data or to reveal hidden data. An attack may use brute force, where every key is tried, or a sophisticated approach such as differential cryptanalysis. An attacker may use only known ciphertext or known ciphertext and plaintext. (2) An attempt to break into a computer or a network or to hamper their operations.

Audit Trail. A record of all of a computer's activities during a certain time period. A trail is produced automatically by an operating system routine or a special utility and is saved as a log file. It can later be used by administrators or security experts to identify improper or unauthorized use of the computer.

Auditing. The process of collecting and analyzing information in order to ensure a proper level of security, as well as compliance with the policies of an organization.

Authentication. The process of verifying a user's identity or authority. Alternatively, the process of establishing the validity of a message. (See also Biometrics.)

Authorization. The process of empowering someone to perform an operation or to have access to restricted resources.

Availability. A computing resource (such as a file server of an organization) should be available to legitimate users. It often happens that malicious persons attack the availability of a resource thereby making it unusable without damaging the resource itself. (See also DoS, DDoS.)

Backdoor. A hidden feature in a piece of software that gives certain people special privileges denied to others. A typical example is a backdoor placed in an encryption algorithm by its author. The author can use the backdoor to decrypt messages without knowing the encryption key. In 1997 the American Senate approved a bill that would have banned the manufacture, distribution, or import of any encryption product that did not include a backdoor for the federal government, but that bill never became a law.

Backdoor Trojan. A Trojan horse that enables a remote user to access and control a computer. This constitutes unauthorized access.

Bacterium. Another name for a computer virus that's not a rabbit. (See also Virus, Rabbit.)

Backup. The process of creating a true copy of a set of data files.

Bandwidth. The capacity of a communications channel. Measured in amount of data per unit time, such as bits per second (baud).

Bayesian filtering. A statistical method that determines whether email is spam. It is based on Bayesian probability theory that computes the probability of an event A given that another event B has occurred.

Biometrics. Identifying or authenticating a person by checking certain physical characteristics such as fingerprints or eye and facial features. (See also Authentication.)

BIOS. An acronym that stands for Basic Input/Output System. BIOS is the lowest level of the operating system routines that control input/output operations. It interfaces directly with hardware.

Bitrate. Bits per second. A measure of the speed of a process such as encrypting or decrypting a file.

Blackhole list. A published list, usually commercial, of addresses known to be sources of spam. (See also Real-time blackhole list.)

Blacklist. A list of email addresses and domains from which no email will be accepted. Used by firewalls and email filters.

Block cipher. A symmetric cipher that encrypts a message by breaking it down into blocks and encrypting each block separately. DES, IDEA, and AES are block ciphers.

Boot sector. The part of the operating system that is first read into memory from disk when a computer is turned on or restarted. The program in the boot sector is then executed, which in turn loads the rest of the operating system. (See also Booting.)

Boot sector virus. A virus that resides in the boot sector of a disk.

Booting. The process of turning a computer on. The main task of booting is to load the operating system from disk. (See also Boot sector.)

Browser. A computer program that locates a Web site (a server), downloads data from it in html format, and displays it as text and graphics. (See also Web browser.)

Brute-force attack. An attempt to break an encrypted message by trying every possible key.

Buffer overflow. An unusual situation that occurs when a program tries to store data past the end of a buffer (an array). Such data overwrites the instructions or data located past the array, and so may cause unexpected results. This is a common technique exploited by hackers to corrupt or infect executable code. To solve such a problem, the program has to check every index used to store data in the array and make sure indexes never point outside the array.

Bug. An error in the design or implementation of a computer program.

Byte. A set of eight bits. This is often the smallest addressable unit in a computer's memory. The number 8 was chosen because one character (ASCII) code or two decimal digits can be stored in 8 bits. (See also ASCII.)

Caesar cipher. A cipher where each letter is replaced by the letter located cyclically n positions in front of it in the alphabet. (See also Affine cipher.)

CGI. An acronym for Common Gateway Interface. A standard employed by a Web server to run programs or scripts and send the output to a user's Web browser.

Checksum. The result of a computation that involves all the bits of a piece of data (a file or a message). The checksum is later used to verify the validity of the data, because virtually any modification of the data will change its checksum.

Cipher. An encryption algorithm that depends on a key.

Ciphertext. The encrypted result produced by a cipher. (See also Plaintext.)

Client. Software that requests and uses a service provided by another program (a server). Often, the server may itself be a client of some other server. (See also Server.)

Code. A symbol that represents another symbol (also a set of symbols that represent other symbols). The ASCII code, for example, represents a set of 128 characters by a set of 128 8-bit codes.

Code (in cryptography). A cryptographic technique that uses a codebook to replace words and letters in the plaintext with symbols from the codebook.

Companion virus. A virus that exploits a feature in certain operating systems that allows for two programs with the same name but different extensions. The operating system uses the file extension to decide which program to execute.

Complex dictionary checking. A feature of anti-spam software that locates (in a dictionary) words often used in spam, even if letters are replaced with lookalike numerals or characters (such as "1nterest r@te").

Computer Emergency Response Team (CERT). An organization that responds to attacks on computers and networks. CERT publishes alerts concerning vulnerabilities and threats, and offers other information to help improve computer and network security.

Computer Network. See Network.

Cookie. A small amount of data that stores information in a computer with the user's permission. Cookies are normally used to enable a Web site to track visits and remember visitors' information.

Corruption. An accidental or intentional modification of computer programs or data.

Covert channels. Physical means by which information is sent between two parties secretly using normal network and computing procedures.

Cryptanalysis. The science and art of breaking encryption (recovering plaintext from ciphertext when the key is unknown). (See also Attack.)

Cryptanalyst. One who tries to break encrypted codes.

Cryptographer. One who develops encryption methods.

Cryptography. The art and science of using mathematics to obscure the meaning of data by applying transformations to the data that are impractical or impossible to reverse without the knowledge of some key. The term comes from the Greek for "hidden writing."

Cryptology. The branch of mathematics concerned with secret writing in all its forms. It includes cryptography, cryptanalysis, and steganography.

CSV. An acronym for Comma Separated Values. CSV is a file format where values (for example, the values from an Excel spreadsheet) are displayed separated by commas. The format does not support macros, so that it cannot spread macro viruses.

Daemon. An operating system routine that runs continuously and forwards input/output requests to other programs or processes as appropriate. The term daemon originated in Unix. The Windows operating system refers to daemons as system agents and services.

Data diddling. Alteration of data. This term refers to what a malicious virus may do to data files it locates in an infected computer.

Data encryption standard (DES). A block cipher based on the work of Horst Feistel in the 1970s that is widely used in commercial systems. DES is a 64-bit block cipher with a 56-bit key organized in 16 rounds of operations.

Data leakage. The theft of data (including software).

DDoS. See Distributed denial of service.

Decryption. The process of converting ciphertext back to plaintext by means of a key. The inverse of encryption. (See also Ciphertext, Encryption, Plaintext.)

Denial of service attack. An attempt to prevent the use of a Web server by sending a vast number of simultaneous messages or requests.

Dictionary attack. Brute-force software that bombards a mail server with email addresses that are generated alphabetically, looking for valid addresses. The same method can be used to guess passwords.

Diffie–Hellman (DH). A public-key cryptography algorithm that generates a shared secret key between two entities after they publicly share some randomly-generated data.

Digital. An approach where all types of data—text, images, audio, and video—are represented in terms of digits (normally bits).

Digital signature. Data value generated by a public-key algorithm based on the content of a block of data and on a private key. It generates an individualized checksum.

Digram. A pair of consecutive symbols.

Disassembly. The process of translating a program in machine language to assembler language.

Disaster-recovery plan (DRP). A procedure developed and periodically rehearsed and revised to ensure quick and complete recovery of an organization from various disasters.

Distributed denial of service. A denial of service attack coming from many computers. (See also Denial of service.)

Domain hijacking. An attack where a hacker takes over a domain by first blocking access to the domain's name server and then replacing it with his own name server.

DoS. See Denial of service.

Downloading. The transfer of data into one's computer. The opposite of uploading. (See also Uploading.)

Dumpster diving. Obtaining private and personal data by searching through discarded documents, disks, and other media. (See also Scavenging.)

DVD. An optical disc, similar to a CD but with seven times the data capacity. A DVD can have 1, 2, or 4 tracks (or layers), with capacities of up to 17.08 Gb. The acronym may either refer to "digital video disc" or "digital versatile disc," or may stand for nothing.

Eavesdropping. Unauthorized interception of data being transmitted.

Electronic fund transfer (EFT). A computerized transaction that can quickly and securely transfer funds electronically between organizations without the need to fill out paper documents.

Emanations analysis. Spying on computer operations by collecting and analyzing signals that are emitted by hardware components.

Encryption. The process of converting plaintext back to ciphertext by means of a key. The inverse of decryption. (See also Ciphertext, Decryption, Plaintext.)

Ethernet. A technology for avoiding message collisions in a local area network (LAN). The ethernet standard is IEEE 802.3.

Eve. A term used in cryptography discussions and examples for the ubiquitous eavesdropper.

Exclusive-OR (XOR). A logical (Boolean) operation that is also its own inverse, which makes it useful in cryptography. It is identical to adding two bits modulo 2. (See also XOR.)

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

Factor. Given an integer N, a factor is any integer that divides it without a remainder.

Factoring. The process of finding the prime factors of an integer.

False positive. A report about a virus or a source of spam that turns out to be wrong.

File infector. A virus that infects executable files and runs each time an infected file is executed. (See also Parasitic virus.)

File server. A computer where data is stored that can be downloaded by authorized computers. (See also Client, Server.)

Firewall. Security software that is placed between the Internet and an organization's local network, or between a network and a computer. The firewall software is governed by rules and passes only network traffic authorized by the rules.

Gateway. A computer that either serves for the transfer of data (for example, a mail gateway that handles all the mail coming into an organization), or a computer that converts data from one protocol to another.

Giga. The quantity giga is defined as $2^{30} = 1{,}073{,}741{,}824$. In contrast, a billion is defined (in the United States) as 10^9. (See Mega.)

Greylist. Email senders who are not blacklisted (excluded) or whitelisted (accepted) can be placed on a greylist and requested to prove that they are sending legitimate mail.

Hacker. Someone who tries to break into computers. A more lenient term is "a computer enthusiast."

Ham. Email that a recipient believes isn't spam.

Harvesting. Scanning the internet for email addresses that can be added to spammers' mailing lists.

Heuristic scanner. A program that detects viruses by using general rules about what viruses are like or how they behave. Conventional anti-virus software looks for known signatures of viruses and is therefore much slower a heuristic scanner. (See also Anti-virus software.)

Hoax. A report about viruses or other security threats, often spread by email, that is intended to deceive. (See also False positive.)

Honeypot. A computer on the internet that used specifically to attract and trap spammers and hackers.

HTML. An acronym for Hypertext Markup Language. The standard for text and images on a Web site.

HTTP. An acronym for Hypertext Transport Protocol. A protocol used by Web servers and clients (browsers) to transfer data to Web browsers.

HTTP scanning. Real-time scanning of HTTP traffic for viruses.

Hypertext. Text that has links to other texts and images.

Integrity. The correctness of a piece of data. An attack on integrity tries to damage data by changing bits. This is why a checksum, normally in the form of CRC, is important.

Internet. THE network that connects many networks and computers all over the world.

Internet protocol (IP). A set of rules governing how data is sent from one computer to another on the Internet.

Interrupt. The way the computer responds to urgent or unusual events. Interrupts involve both hardware and software.

IP Address. A unique 32-bit number assigned to each computer on the Internet. It is used as the unique address of the computer by Internet protocols. There can be 2^{32} IP addresses (about four billion).

ISO. The International Standards Organization. This is one of the organizations responsible for developing standards. Among other things, it is responsible (together with the ITU) for the JPEG and MPEG compression standards. (See also ITU.)

ITU. The International Telecommunications Union, the new name of the CCITT, is a United Nations organization responsible for developing and recommending standards for data communications.

Java. A platform-independent higher-level programming language designed specifically for the Web. Programs written in Java are either applications or applets. (See also Java applet, Sandbox.)

Java applet. A small application normally used to display text and graphics on Web pages. Applets are run by the browser in a safe environment and cannot make changes to the client's computer. (See also Java, Sandbox.)

JFIF. An acronym for JPEG File Interchange Format. JFIF is a graphics file format that makes it possible to exchange JPEG-compressed images between different computers. The main features of JFIF are the use of the YCbCr triple-component color space for color images (only one component for grayscale images) and the use of markers to specify features missing from JPEG, such as image resolution, aspect ratio, and features that are application specific.

JPEG. A sophisticated lossy compression method [Salomon 04] for color or grayscale still images (not movies). It also works best on continuous-tone images, where adjacent pixels have similar colors.

The main idea behind JPEG is that an image exists for people to look at, so when the image is compressed, it is acceptable to lose image features to which the human eye is not sensitive.

The term JPEG is an acronym that stands for Joint Photographic Experts Group. This was a joint effort by the CCITT and the ISO that started in June 1987. The JPEG standard has proved successful and has become widely used for image presentation, especially in Web pages.

Kerckhoffs's principle. An important principle in cryptography (Section 12.2). It states that the security of an encrypted message must depend on keeping the key secret and should not depend on keeping the encryption algorithm secret.

Key (cryptographic). A string of bits used to encrypt and decrypt messages. In non-computer cryptography the key is a string of any symbols. (See also Key distribution.)

Key distribution. The process of distributing a secret cryptographic key to all the locations of an organization. [See also Key (cryptographic).]

Key space. The number of possible key values. For example, there are 2^{64} key values for a 64-bit key. (See Exercise 12.1.)

Leet. 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 "sploitz" (for exploits).

Link virus. A virus that corrupts directory entries so that they point to the virus file, allowing it to execute when the user types the name of a legitimate application.

Logic bomb. Malicious software, normally a Trojan horse, left in a disk or inside another file to be triggered by a certain event. A disgruntled employee about to be sacked can plant such a bomb in a central file server, waiting to damage files when the employee's id number is deleted from the list of employees. (See also Rogue Software, Time Bomb.)

Macro. A set of instructions and/or data that's assigned a name. When the user types the name, the macro is expanded. Certain applications, such as Microsoft Word and Excel, support a macro facility.

Macro virus. A virus disguised as a macro and infecting data files.

Mail drop. An email address set up by a spammer specifically to receive responses to spam. The spammer opens and closes such accounts frequently.

Malicious software. See Rogue Software.

Malware. See Rogue Software.

Mantrap. A device to prevent unauthorized access to a room without employing a guard. A small booth between two doors where a door can open only when the other door is closed.

Master boot record. The boot sector on a bootable disk. Also known as the partition sector. The first sector that's read and executed when a computer is booted or is restarted.

Mega. Mega is defined as $2^{20} = 1{,}048{,}576$. In contrast, a million is defined as 10^6. (See Giga.)

Memory-resident virus. A virus that copies itself in memory when it is first executed. It modifies certain interrupt handling routines, so it is executed each time any of the routines is invoked.

Modem. An acronym that stands for MOdulator/DEModulator. Modem is hardware that converts data (bits) between computer form and a form that can propagate through telephone lines, radio or satellite link.

Monoalphabetic substitution cipher. A cryptographic algorithm with a fixed substitution rule.

Multipartite virus. A virus that infects both boot sectors and executable files.

Munging. Disguising email addresses so that they cannot be harvested. Recipients are told how (or use their intelligence) to decode the address.

National Computer Security Center (NCSC). A United States government organization that evaluates computing equipment for high-security applications.

National Institute of Standards and Technology (NIST). An agency of the United States government that establishes national standards.

National Security Agency (NSA). A branch of the United States Department of Defense responsible for intercepting foreign communications and for ensuring the security of United States government communications.

Network. A set of computers or computer installations connected by communication channels.

Newsgroup. An electronic forum where users post articles, questions, and followup messages on specific topics.

Obfuscation. A term that refers to (1) disguising email addresses so that spammers cannot harvest them and (2) spammers' attempts to hide messages so that they will not be detected.

One-time pad. An encryption method that employs a large key (as long as the message) to securely encrypt and decrypt a single message. Each encrypted message has to use a fresh key.

Open relay. An SMTP email server that allows the third-party relay of email messages. Spammers and other hackers can hijack such servers and use them to send spam and malicious software.

Operating system. A set of programs that provide important services to the user. In a multiuser computer, the operating system also supervises users. The most common services an operating system provides are file handling (display, save, rename, move, and delete), data handling (editing text and compiling programs), and input/output (high-level routines that handle interrupts and simplify the transfer of data).

Packet. Long messages transmitted over a network are broken up into small chunks called packets (or data packets). This is why a computer network is often referred to as a packet-switching network. The advantage of packets is reliability. If one packet is lost on its way or arrives garbled, only that packet has to be resent. All the packets of a long message contain the same destination address, same identification number, and individual serial numbers. The serial number are used to combine the packets into one message at the destination.

Parasitic virus. See File infector.

Password. A string of symbols (normally letters, digits, and certain punctuation marks) used to identify an authorized computer user. It is important to select strong passwords, keep them secret, and change them periodically.

Password sniffing. Wiretapping a network in order to harvest passwords.

Patch. An update released by a software maker to eliminate bugs and security holes in existing programs.

Phishing. Tricking users into submitting confidential information or passwords by creating a replica of a legitimate Web site or by social engineering methods.

Phreaking. Hacking telephones. Manipulating the way telephones work to void paying for telephone use.

Piggybacking. Sneaking into a restricted facility by following someone while a door is open. Same as Tailgating.

Plaintext. An as-yet unencrypted message. (See also Ciphertext.)

Polyalphabetic substitution. A cryptographic technique where the rule of substitution changes all the time.

Polymorphic virus. Self-modifying virus that changes its code in an attempt to make itself harder to detect. (See also Virus.)

Port. A port is similar to a door in that accessing a network opens up a port in the computer. Each packet of data that arrives at the computer has a port number and certain ports are dedicated to certain network protocols. A port can be thought of as an integer that identifies the endpoint of a communications channel. Once a port is opened on a computer, only one process can listen on it for input.

Port scan. Each port is associated with a process (a program) that listens for input arriving to the port from the outside. Imagine a hacker who discovers a weakness in a certain program that's used to listen to port P. The hacker may decide to send probing messages to port P in all the computers whose IP numbers are in a certain interval. When a computer responds, the hacker adds its IP to the list of potential victims that can later be attacked.

Program. A set of instructions that specifies actions a computer should perform. A program is normally written in a higher-level language and is translated by a compiler into a set of machine instructions. (See also Software.)

Program virus. See File infector.

Protocol. A set of rules, often to standardize procedures for computer communications.

Proxy serve. A server that makes requests to the Internet on behalf of another computer. It sits between a local network and the internet and can be used for security purposes.

Public-key algorithm. A cipher that uses a pair of keys, a public key and a private key, for encryption and decryption. Also called an asymmetric algorithm. (See also Asymmetric algorithm.)

Public-key cryptography. Cryptography based on methods involving a public key and a private key.

Public-key cryptography standards (PKCS). Standards published by RSA Data Security that describe how to use public-key cryptography in a reliable, secure, and interoperable fashion.

Rabbit. A computer virus that does not attach itself to another piece of software and does its damage by monopolizing some computing resource, such as CPU time, memory, or disk space. (See also Bacterium, Virus.)

RAM. Acronym for Random Access Memory, but a misnomer. RAM is really read/write memory. Currently, most computer memories are of this type, which is volatile. It loses its content when power is turned off. (See also ROM.)

Real-time blackhole list (RBL). A list that rejects all email, valid or not, from addresses that are blacklisted because they are known to send spam or to host spammers. Such a list can be employed by ISPs to take anti-spam measures and thereby greatly help their users. (See also Blackhole list.)

Reverse DNS check. Checking an email's sender address against the database of a domain name server to ensure that it originated from a valid domain name or Web address.

Rogue software. A computer program specifically written to damage computing resources. (See also Malicious software, Malware, Virus, Worm, Trojan Horse, Logic Bomb.)

ROM. Acronym for Read Only Memory. ROM is nonvolatile and is therefore used to store permanent data such as the bootstrap loader, Section 2.6. (See also RAM.)

Root kit. A program that's specially 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.

Router. A hardware device that receives messages for computers in a network and forwards them to the individual computers in the network based upon IP addresses.

RSA Data Security, Inc. (RSADSI). A company [RSA 04] primarily engaged in selling and licensing public-key cryptography for commercial purposes.

Sandbox. A mechanism for executing programs in a controlled environment, often used with Java applets. (See also Java.)

Scavenging. Probing a computer (or even discarded old disks) at random for data useful to a hacker. (See also Dumpster diving.)

Secure socket layer (SSL). A protocol enabling the secure transfer of sensitive information on the Internet. The sensitive data is encrypted by a block cipher, and the SSL protocol is used to select a random key for each transfer and communicate it securely through unsecured channels.

Security (computer). The field that has to do with guaranteeing the availability, confidentiality, and integrity of computing systems.

Server. A program that provides data in response to requests from other programs called clients. If a computer is dedicated to running servers, it is also called a server. (See also Client, File server.)

Session. A process where an entire network protocol is executed between two computers (hosts).

Session hijacking. The process of taking over a session that someone else has started.

SHS. A 3-letter file extension for Windows "scrap object" files. These files can include virtually any code and execute when clicked on. The extension itself may be hidden.

Smart card. A plastic card that includes a chip. The chip is either a microprocessor or memory. The smart card authenticates its owner and permits certain transactions such as using a pay telephone or public transportation, or withdrawing money from an ATM.

SMTP. An acronym for Simple Mail Transport Protocol. The protocol for delivering Internet email.

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

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 they type their password, or by sending email that poses as an official notice asking for sensitive information. Even though no special software may be needed and no software weakness is exploited, this is still a tool used by many.

Software. A set of instructions (in assembler) or statements (in a higher-level language) that carry out a task on the computer. Computers are useful because the same computer can execute many programs and thus perform many different tasks. However, without a program, a computer can do nothing. (See also Program.)

Spam. Commercial and bulk email sent unsolicited and in large quantities in an attempt to trap a small percentage of the receivers into buying useless products and services.

Spambot. Software used by spammers to find and harvest email addresses from the Internet.

Spoofing. The term spoof means to pretend to be someone else. Spoofing is forging the sender's address in email. It is used mostly to hide the origin of spam, or to convince recipients that the email came from a familiar or reliable source.

Spyware. Software that tracks user activity without the user's knowledge and reports this information to its "owner."

Surge suppressor. See UPS.

Tarpitting. Any technique to monitor email in order to discover sources of large quantities of email that may be spam.

Tarpit. An email server that's kept intentionally slow in order to trap spammers that use harvesting robots.

TCP/IP. Acronyms for Transmission Control Protocol/Internet Protocol. The collective name for the two chief Internet protocols.

Threat. A potential for a security violation. A threat exists when someone discovers a security weakness and attempts to exploit it for harmful purposes.

Time Bomb. A logic bomb triggered at a certain point in time. (See also Logic Bomb, Rogue Software.)

Trapdoor. See Back door.

Trojan horse. Malicious (rogue) software that hides itself in the computer in an attempt to harm. A typical Trojan horse may collect keystrokes and transmit them to its owner who may be looking for passwords and other personal information typed by the user. (See also Rogue Software.)

Trust. The process of determining who gets what permissions and who can perform certain actions on a computer.

Unicode. A new international standard code, the Unicode, has been proposed, and is being developed by the international Unicode organization (`www.unicode.org`). Unicode uses 16-bit codes for its characters, so it provides for $2^{16} = 64\text{K} = 65{,}536$ codes. (Notice that doubling the size of a code much more than doubles the number of possible codes. In fact, it *squares* the number of codes.) Unicode includes all the ASCII codes in addition to codes for characters in foreign languages (including complete sets of Korean, Japanese, and Chinese characters) and many mathematical and other symbols. Currently, about 39,000 out of the 65,536 possible codes have been assigned, so there is room for adding more symbols in the future. (See also ASCII.)

Uninterrupted Power Supply (UPS). A device that "cleans" the power supplied by the power grid. A UPS employs special circuits to suppress power surges and uses a battery to temporarily supply electrical power when the voltage drops.

Unix. A popular multiuser, multitasking operating system that originated at Bell Labs in the late 1960s by a handful of programmers. Unix was originally envisioned as a small, flexible operating system used exclusively by programmers, but has developed over the years in response to changing demands and technological innovations. Today, Unix is the operating system of choice of many unsophisticated computer users.

Uploading. The transfer of data from one's computer. The opposite of downloading. (See also Downloading.)

URL. An acronym for Uniform Resource Locator. A Web address.

User. A person, an organization, or a process that accesses a computer. A user can be authorized or not.

VBS. Acronym for Visual Basic Script. VBS is executable code embedded in an application, document, or a Web page that can run as soon as the page is viewed.

Vernam cipher. Cipher developed for encrypting teletype traffic by computing the exclusive OR of the data bits and the key bits. This is a common approach to constructing stream ciphers. (See One-time pad.)

Virus. Malicious (rogue) software that infects other programs. In practice, viruses tend to harm the computer they are in, and also replicate themselves and send copies outside. (See also Rogue Software.)

Virus identity. A detailed description of virus features used by anti-virus software for virus recognition.

Virus scanner. Anti-virus software. Most scanners are virus-specific, they identify and delete only viruses that are already known. (See also Anti-virus software, Heuristic scanner.)

Vulnerability. A flaw or weakness in the design, implementation, or operation of a piece of hardware or software that could be exploited to violate security.

Vulnerability scanner. A program especially designed to quickly check computers on a network for known weaknesses. A port scanner is a special case. It's 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.

WAP: Wireless Application Protocol. Internet-type protocol that provides information to mobile telephones.

Web. See World wide web.

Web browser. Client software to access and display the html content of Web sites. The HTTP protocol is used to transfer html documents. (See also Browser.)

Web bug. A small image inserted in an email or Web page that alerts a spammer when a message is read or previewed.

Web server. A computer connected to the Internet that stores a Web site in html format and can make it accessible with the HTTP protocol.

Whitelist. A list of trusted email addresses from which email is accepted without checking it for spam and/or viruses.

Wiretapping. Intercepting data as it moves along a communications channel.

Workstation. A single-user computer, often connected to a network. Nowadays, there is no difference between a workstation and a personal computer, but in the past workstations were more powerful.

World Wide Web. The collection of Web servers all over the world.

Worm. Rogue software that replicates and transmits copies of itself through a network. A worm may damage its host, or is designed to use the host's computing resources for the benefit of its owner. (See also Rogue Software.)

WWW. See World Wide Web.

XOR. See Exclusive OR.

Zombie. A computer that has been hijacked and is under the remote control of a hacker. Zombies are used to send spam or launch a denial of service attack.

Bibliography

3M (2004) is http://cms.3m.com/cms/US/en/2-68/iclcrFR/view.jhtml.

absolute (2005) is URL http://www.absolute.com/public/main/.

ACA (2005) is URL http://www.und.nodak.edu/org/crypto/crypto/.

Aegean Park Press (2001) is URL http://www.aegeanparkpress.com/.

AFAC (2005) is URL http://www-vips.icn.gov.ru/.

Agrawal, Rakesh, and Ramakrishnan Sirkant (2004) "Privacy-Preserving Data Mining," available from
http://www.almaden.ibm.com/software/quest/Publications/papers/sigmod00˙privacy.pdf.

Akamai (2004) is www.akamai.com.

Amiga (2004) is URL http://www.amiga.org/.

Anderson, Ross, Roger Needham, and Adi Shamir (1998) "The Steganographic File System," in David Aucsmith (ed.) *Proceedings of the Second Information Hiding Workshop, IWIH*, pp. 73–82, April. Also available from URL
http://citeseer.nj.nec.com/anderson98steganographic.html.

anonymizer (2005) is URL www.anonymizer.com.

APWG (2004) is http://www.antiphishing.org/.

arin (2004) is http://www.arin.net/whois/.

Arnold, Michael, Martin Schmucker, and Stephen D. Wolthusen (2003) *Techniques and Applications of Digital Watermarking and Content Protection*, Boston, Artech House.

Asonov, Dmitri and Rakesh Agrawal (2004) "Keyboard Acoustic Emanations," *IEEE Symposium on Security and Privacy*, Oakland, California, pp. 3–11, May. Available at http://www.almaden.ibm.com/software/quest/Publications/papers/ssp04.pdf.

attrition (2004) is URL http://www.attrition.org/.

attrition-mirror (2005) is URL http://www.attrition.org/mirror/attrition/.

Aura, Tuomas (1996) "Practical Invisibility in Digital Communication," in *Proceedings of the Workshop on Information Hiding*, Cambridge, England, May 1996, pp. 265–278, *Lecture Notes in Computer Science* **1174**, New York, Springer Verlag. Also available from URL http://www.tcs.hut.fi/Personnel/tuomas.html.

avenger (2005) is http://www.research.ibm.com/antivirus/SciPapers/Gordon/Avenger.html.

badguys (2005) is URL http://www.badguys.org/.

Bamford, James (2002) *Body of Secrets: Anatomy of the Ultra-Secret National Security Agency*, New York, Anchor books (Random House).

Barker, Wayne G. (1984) *Cryptanalysis of Shift-Register Generated Stream Cipher Systems*, Laguna Hills, Calif., Aegean Park Press, vol. **C-39**.

Barker, Wayne G. (1989) *Introduction to the Analysis of the Data Encryption Standard (DES)*, Laguna Hills, Calif., Aegean Park Press, vol. **C-55**.

Barker, Wayne G. (1992) *Cryptanalysis of the Single Columnar Transposition Cipher*, Laguna Hills, Calif., Aegean Park Press, vol. **C-59**.

Bauer, Friedrich Ludwig (2002) *Decrypted Secrets: Methods and Maxims of Cryptology* 3rd edition, Berlin, Springer Verlag.

bbbseal (2005) is URL www.bbbonline.org.

bbc (2004) is http://www.bbc.co.uk/dna/h2g2/A787917.

Bell, D. E., and L. J. LaPadula (1974) "Secure Computer Systems: Mathematical Foundations and Model," Technical report, MITRE.

Blakley, G. R. (1979) "Safeguarding Cryptographic Keys," in *AFIPS Conference Proceedings*, **48**:313–317.

Blowfish (2005) is URL http://www.schneier.com/blowfish.html.

Boneh D., and D. Brumley (2004) "Remote Timing Attacks Are Practical," available from http://crypto.stanford.edu/%7Edabo/abstracts/ssl-timing.html.

BPCS (2003) is URL http://www.know.comp.kyutech.ac.jp/BPCSe/ file BPCSe-principle.html.

Brenner, Susan W. (2002) "Organized Cybercrime? How Cyberspace May Affect the Structure of Criminal Relationships," *North Carolina Journal of Law and Technology*, **4**(1).

business.com (2004) is http://www.business.com/directory/computers_and_software/security/hardware/tracking_and_theft_prevention/.

CA (2004) is http://www3.ca.com/securityadvisor/virusinfo/default.aspx.

Calif-gov (2005) "Your Social Security Number: Controlling the Key to Identity Theft," available online at URL `http://www.privacy.ca.gov/sheets/cis4english.htm`.

Campbell, K. W., and M. J. Wiener (1993) "DES Is Not a Group," *Advances in Cryptology, CRYPTO '92*, New York, Springer Verlag, pp. 512–520.

Casanova, Giacomo (1757) *Histoire de Ma Vie*, in 12 volumes. Translated by Willard R. Trask as *The History of My Life*, Baltimore, Johns Hopkins University Press, 1967, reissued 1997.

CDC (2004) is `http://www.cdc.gov/`.

CERT (2004) is URL `http://www.cert.org/other_sources/viruses.html`.

chatdanger (2005) is URL `http://www.safekids.com/chatdanger.htm`.

Chomsky, Noam, and George A. Miller (1958) "Finite State Languages," *Information and Control*, **1**(2)91–112, May.

Code Red II (2001) "Code Red II: Another Worm Exploiting Buffer Overflow In IIS Indexing Service DLL," CERT Incident Note IN-2001-09, Aug. 6. Available online at `http://www.cert.org/incident_notes/IN-2001-09.html`.

Cohen, Fred (1987) "A cryptographic checksum for integrity protection,' *Computers and Security*, **6**(6)505–510, December 1.

Cohen, Frederick B. (1994a) *A Short Course on Computer Viruses*, 2nd edition, New York, NY, John Wiley.

Cohen, Frederick B. (1994b) *It's Alive! The New Breed of Living Computer Programs*, New York, NY, John Wiley.

comscore (2004) is URL `http://www.comscore.com/`.

Conceptlabs (2004) is URL `http://www.conceptlabs.co.uk/alicebob.html`.

cookiecentral (2004) is `http://www.cookiecentral.com/faq/#2.9`.

Coppersmith, Donald, and Philip Rogaway (1994) "A Software-Optimized Encryption Algorithm," *Fast Software Encryption, Cambridge Security Workshop Proceedings*, New York, Springer-Verlag, pp. 56–63.

Coppersmith, Donald, and Philip Rogaway (1995) "Software-Efficient Pseudorandom Function and the Use Thereof for Encryption," United States Patent 5,454,039, 26 September.

corallosoftware (2005) is `http://www.corallosoftware.com/index.html`.

Cox, Ingemar J. (2002) *Digital Watermarking*, San Francisco, Morgan Kaufmann.

Crap (2005) is URL `http://www.mat.dtu.dk/people/Lars.R.Knudsen/crap.html`.

creditexpert (2005) is URL `https://www.creditexpert.com/`.

creditreporting (2005) is URL `http://affiliates.creditreporting.com/`.

Cryptologia (2005) is URL http://www.dean.usma.edu/math/pubs/cryptologia/.

Cryptology (2005) is URL http://link.springer.de/link/service/journals/00145/.

CSE (2005) is URL http://www.cse.dnd.ca/.

csrc (2004) is http://csrc.nist.gov/CryptoToolkit/tkhash.html.

CVE (2001) is CVE-2001-0500, *Buffer overflow in ISAPI extension*, available online at http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2001-0500.

cyberpatrol (2005) is URL http://www.cyberpatrol.com/.

CyberSitter (2005) is URL http://www.cybersitter.com/.

cypherpunks (2004) is ftp://ftp.csua.berkeley.edu/pub/cypherpunks/steganography/.

Dawkins, Richard (1990) *The Selfish Gene*, 2nd Edition, New York, Oxford University Press.

Day (2004a) is http://www.csse.uwa.edu.au/~pd/securing_mac_os_x.pdf.

Day (2004b) is http://www.csse.uwa.edu.au/~pd/securing_mac_os_x_present.pdf.

Denning, Peter J. (1990) *Computers Under Attack: Intruders, Worms, and Viruses*, New York, ACM Press and Addison Wesley.

DES2 (1993) is http://www.itl.nist.gov/fipspubs/fip46-2.htm.

DES3 (1999) is csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf.

digitalenvoy (2004) is www.digitalenvoy.net/.

dodgeit (2004) is http://www.dodgeit.com/.

DSD (2005) is URL http://www.dsd.gov.au/.

dslreports (2004) is URL http://www.dslreports.com/scan.

Duke (2005) is URL http://www.oit.duke.edu/helpdesk/filesharing/.

Dunham W. (1990) *Journey Through Genius: The Great Theorems of Mathematics*, New York, John Wiley.

dvdbook (2005) is URL http://www.dvdforum.org/tech-dvdbook.htm.

eBates (2005) is URL https://www.ebates.com/.

eeggs (2005) is URL http://www.eeggs.com/.

EICAR (2004) is URL http://www.eicar.org/.

Encyc1 (2004) is http://www3.ca.com/securityadvisor/virusinfo/browse.aspx.

Encyc2 (2004) is http://securityresponse.symantec.com/avcenter/vinfodb.html.

ensuretech (2004) is URL http://www.ensuretech.com/.

equifax (2005) is URL http://www.equifax.com/.

Feige, Uriel, Amos Fiat, and Adi Shamir (1988) "Zero Knowledge Proofs of Identity," *Journal of Cryptology*, **1**(2)77–94.

Feistel, Horst (1973) "Cryptography and Computer Privacy," *Scientific American*, **228**(5) 15–23, May.

FIPS-180 (2005) is `http://www.itl.nist.gov/fipspubs/fip180-1.htm`.

FIPS-185 (2005) is `http://www.itl.nist.gov/fipspubs/fip185.htm`.

FIPS-186 (2005) is `http://www.itl.nist.gov/fipspubs/fip186.htm`.

Flannery, Sarah, and David Flannery (2001) *In Code: A Mathematical Journey*, Workman Publishing Company.

FreeBSD Words (2005) is URL `ftp://www.freebsd.org/usr/share/dict/words`.

freedom (2005) is URL `www.freedom.net`.

Friedman, William F. (1996) *The Index of Coincidence and Its Applications in Cryptanalysis*, Laguna Hills, Calif., Aegean Park Press, vol. **C-49**.

f-secure (2005) is URL `http://www.f-secure.com/`.

ftc (2004) is URL `http://www.ftc.gov/opa/2003/09/idtheft.htm`.

FTC-CONT (2005) is URL `www.ftc.gov/bcp/conline/pubs/online/sitesee.html`.

FTC-infosecurity (2005) is URL `http://www.ftc.gov/bcp/conline/edcams/infosecurity/index.html`.

FTC-privacy (2005) is `http://www.ftc.gov/privacy/index.html`.

FTC-work (2005) is `http://www.ftc.gov/bcp/workshops/spyware/index.htm`.

fundrace (2005) is URL `http://fundrace.org/citymap.php`.

Gaines, Helen Fouché (1956) *Cryptanalysis: A Study of Ciphers and Their Solutions*, New York, Dover.

Garfinkel, Simson (1995) *PGP: Pretty Good Privacy*, Sebastopol, Calif., O'Reilly.

GCHQ (2003) is URL `http://www.gchq.gov.uk/`.

gemplus (2005) is URL `http://www.gemplus.com/`.

Gerrold, David (1988) *When HARLIE Was One*, Bantam Spectra (Random House), Updated edition.

getnetwise (2005) is URL `www.getnetwise.org/`.

getnetwise-ctrct (2005) is URL `www.getnetwise.org/tools/toolscontracts.php`.

GnuPG (2004) is `http://www.gnupg.org/`.

Golomb, Solomon W. (1982) *Shift Register Sequences*, 2nd edition, Laguna Hills, Calif., Aegean Park Press.

Google (2005) is URL desktop.google.com.

Gordon, Sarah (2005) "Virus Writers: The End of The Innocence?" available at URL http://www.research.ibm.com/antivirus/SciPapers/VB2000SG.htm.

Grampp, F. T., and R. H. Morris (1984) "UNIX Operating System Security," *Bell Laboratories Technical Journal*, **63**(8)1649–1672, October.

gregorybraun (2005) is URL http://www.gregorybraun.com/PassKeep.html.

Guillou, Louis, and Jean-Jacques Quisquater (1988) "A Practical Zero-Knowledge Protocol Fitted to Security Microprocessors Minimizing Both Transmission and Memory," in *Advances in Cryptology, Eurocrypt '88 Proceedings*, pp. 123–128, Berlin, Springer-Verlag.

Guinness (2004) is www.guinnessworldrecords.com/.

Gutenberg (2005) is URL http://www.gutenberg.net/.

Guthke, Karl S. and Robert C. Sprung (1991) *Traven: The Life Behind the Legends*, Chicago, IL, Lawrence Hill Books.

Harley, David, Robert Slade, and Urs Gattiker (2001) *Viruses Revealed*, Berkeley, CA, Osborne/McGraw-Hill.

HastaLaVista (2004) is http://www.hastalavista.com (but don't use it, even at your own risk).

Hinsley, F. H., and Alan Stripp (eds.) (1992) *The Codebreakers: The Inside Story of Bletchley Park*, Oxford, Oxford University Press.

homograph (2005) is the PDF document at http://www.cs.technion.ac.il/~gabr/papers/homograph_full.pdf.

honker (2004) is URL http://www.cnhonker.com/.

House (2004) "Spyware: what you don't know can hurt you." Hearing before the subcommittee on commerce, trade, and consumer protection, April 29, 2004, serial no. 108-89. Available at http://www.access.gpo.gov/congress/house/house05ch108.html.

Hydan (2005) is URL http://www.crazyboy.com/hydan/.

IANA port (2004) is URL http://www.iana.org/assignments/port-numbers.

IbmAntiVirus (2005) http://www.research.ibm.com/antivirus/.

idtheftcenter (2004) is http://www.idtheftcenter.org/index.shtml.

IIS (2004) is www.microsoft.com/iis.

Information Week (2004) is URL http://www.informationweek.com/story/showArticle.jhtml?articleID=52601698.

insecure (2004) is URL http://www.insecure.org/nmap/.

intelius (2005) is URL http://find.intelius.com/.

IOCCC (2004) is http://www.ioccc.org/.

iridiantech (2005) is URL http://www.iridiantech.com/.

ISIS (2005) http://www.imrg.org/8025696F004581B3/pages/imrg+Resources.

ISS (2005) is URL www.iss.net.

ITU (2005) is URL http://www.itu.int/home/index.html.

Jack Moratis (2004) is http://home.comcast.net/~educbc5454/software.html.

Jargon (2004) is URL http://www.catb.org/~esr/jargon/.

Johnson, Neil F., et al. (2001) *Information Hiding: Steganography and Watermarking—Attacks and Countermeasures, Advances in Information Security*, volume 1, Boston, Kluwer Academic.

junkbusters declare (2005) is URL http://www.junkbusters.com/declare.html.

Kahn, David (1996) *The Codebreakers: The Comprehensive History of Secret Communications from Ancient Times to the Internet*, revised edition, New York, Scribner.

Katzenbeisser, Stefan, and Fabien A. P. Petitcolas (eds.) (2000) *Information Hiding Techniques for Steganography and Digital Watermarking*, Norwood, Mass., Artech House.

Kerckhoffs, Auguste (1883) "La Cryptographie Militaire," *Journal des Sciences Militaires*, **9**:5–38, 161–191, January–February. Also available in html format from URL http://www.petitcolas.net/fabien/kerckhoffs/la_cryptographie_militaire_i.htm.

Knowspam (2004) is URL http://www.Knowspam.net/.

knowyourloanrate (2005) is URL https://www.knowyourloanrate.com/.

Knuth, Donald E. (1984) *The TEXBook*, Reading, Mass., Addison-Wesley.

Konheim, Alan G. (1981) *Cryptography: A Primer*, New York, John Wiley and Sons.

Kuhn (2004) is URL http://www.cl.cam.ac.uk/TechReports/UCAM-CL-TR-577.pdf.

Larson, P. Å., and A. Kajla (1984) "Implementation of a Method Guaranteeing Retrieval in One Access," *Communications of the ACM*, **27**(7)670–677, July.

Lavasoft (2004) is http://www.lavasoftusa.com/ or http://www.lavasoft.de/.

Levy, Steven (2002) *Crypto: How the Code Rebels Beat the Government Saving Privacy in the Digital Age*, Penguin Putnam.

Mailblocks (2004) is URL http://www.Mailblocks.com.

MailFrontier (2004) is URL http://www.MailFrontier.com/.

mailinator (2004) is URL http://www.mailinator.net/mailinator/Welcome.do.

Maiwald, Eric and William Sieglein (2002) *Security Planning and Disaster Recovery*, Berkeley, CA, Osborne/McGraw-Hill.

maporama (2004) is http://www.maporama.com/share/.

mapquest (2004) is http://www.mapquest.com/maps/latlong.adp.

MathWorld (2005) is html file Gram-SchmidtOrthonormalization.html in URL http://mathworld.wolfram.com/.

McAfee (2004) is URL http://www.mcafee.com/us/.

McDonald, Andrew D., and Markus G. Kuhn (1999) "StegFS: A Steganographic File System for Linux," in *Proceedings of Information Hiding*, New York, Springer-Verlag, LNCS **1768**, pp. 463–477. Also available from http://www.mcdonald.org.uk/StegFS/.

MD5 (2004) is ftp://ftp.umbc.edu/pub/unix/rfc/rfc1321.txt.gz.

melissavirus (2005) is URL www.melissavirus.com/.

Merkle, R. C., and M. Hellman (1981) "On the Security of Multiple Encryption," *Communications of the ACM*, **24**(7)465–467.

missingkids (2005) is URL http://www.missingkids.com/.

Mitnick, Kevin D. and William Simon (2002) *The Art of Deception: Controlling the Human Element of Security*, New York, John Wiley.

Moore, Dan Tyler and Martha Waller (1965) *Cloak and Cipher*, Indianapolis, IN, Bobbs-Merrill, 1962; London, Harrap.

MS04-028 (2004) is http://www.microsoft.com/technet/security/bulletin/MS04-028.mspx.

MSoffice (2005) is URL http://office.microsoft.com/en-us/officeupdate/.

MSsecurity (2005) is URL http://www.microsoft.com/security/.

MStechnet (2005) is URL http://www.microsoft.com/technet/security/.

MTX (2005) is URL http://www.f-secure.com/v-descs/mtx.shtml.

NCM (2005) is URL http://www.nsa.gov/museum/.

netnanny (2005) is URL http://www.netnanny.com/.

Network solutions (2004) is URL http://www.networksolutions.com/en_US/whois/index.jhtml.

networkusa (2005) is URL http://www.networkusa.org/fingerprint.shtml.

Newton, David E. (1997) *Encyclopedia of Cryptology*, Santa Barbara, Calif., ABC-Clio.

Nicetext (2005) is URL http://www.nicetext.com/.

NIST (1992) "The Digital Signature Standard, proposal and discussion," *Communications of the ACM*, **35**(7):36–54.

NIST (2004) is URL http://csrc.nist.gov/virus/.

NIST Handbook (2004) is available at
csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf.

NISTunits (2004) is http://physics.nist.gov/cuu/Units/binary.html.

NSA (2004) is http://www.nsa.gov/.

NSA-SEC (2005) is URL http://www.nsa.gov/snac/.

NSA-venona (2004) is http://www.nsa.gov/venona/.

onion-router (2005) is URL http://www.onion-router.net/.

OpenPGP (2005) is http://www.openpgp.org/.

opensecrets (2005) is URL http://www.opensecrets.org/indivs/.

OpenSSL (2004) is the OpenSSL project, located at http://www.openssl.org.

Orebaugh, Angela D. and Gilbert Ramirez (2004) *Ethereal Packet Sniffing*, Rockland, Mass., Syngress.

packet-sniffing (2004) is URL http://www.packet-sniffing.com/.

pchell (2005) is URL http://www.pchell.com/virus/mtx.shtml.

PCPhoneHome (2004) is http://www.pcphonehome.com/.

performics (2005) is http://www.performics.com/about/press/code_of_conduct.pdf.

Petitcolas (2003) is URL http://www.petitcolas.net/fabien/steganography/bibliography/.

Pfitzmann, B. (1996) "Information Hiding Terminology," in *Information Hiding*, New York, Springer *Lecture Notes in Computer Science*, **1174**:347–350.

ping (2004) is URL http://ftp.arl.mil/~mike/ping.html.

PKCS (2004) is http://www.rsasecurity.com/rsalabs/node.asp?id=2124.

Pohlmann, Ken (1992) *The Compact Disc Handbook*, 2nd edition, Middleton, Wisconsin, A-R Editions.

privacyalliance (2005) is URL http://www.privacyalliance.org/.

privacyrights (2005) is URL http://www.privacyrights.org/fs/fs21-children.htm.

protectkids (2005) is URL http://www.protectkids.com/.

purityscan (2005) is URL http://www.purityscan.com/.

qspace (2005) is URL http://qspace.iplace.com/.

Quova (2004) is www.quova.com/.

Raymond (2004) is URL http://www.catb.org/~esr/faqs/.

Reiter, Michael K. and Aviel D. Rubin (1998) "Crowds: Anonymity for Web Transactions," *ACM Transactions on Information and System Security*, **1**(1)66–92

remotelyanywhere (2004) is URL http://www.remotelyanywhere.com/.

Rescorla, Eric (2000) *SSL and TLS: Designing and Building Secure Systems*, Reading, Mass., Addison Wesley.

RFC-862 (2004) is at URL www.faqs.org/rfcs/rfc862.html.

RFC-864 (2004) is at URL www.faqs.org/rfcs/rfc864.html.

rfc1321 (2005) is URL http://www.ietf.org/rfc/rfc1321.txt.

RFC-1738 (2004) is at URL www.faqs.org/rfcs/rfc1738.html.

RIAA (2005) is URL http://www.riaa.com/about/members/.

Ritter (2005) is URL http://www.ciphersbyritter.com/ARTS/PRACTLAT.HTM.

Ritter, Terry (1990) "Substitution Cipher with Pseudo-Random Shuffling: The Dynamic Substitution Combiner," *Cryptologia* **14**(4)289–303. An updated version is available at http://www.ciphersbyritter.com/DYNSUB.HTM.

Rivest, R. (1991) "The MD4 Message Digest Algorithm," in Menezes, A. J., and S. A. Vanstone, (eds.), *Advances in Cryptology: CRYPTO '90 Proceedings*, pp. 303–311, New York, Springer-Verlag.

Rivest, R. (1992) "The MD4 Message Digest Algorithm," RFC 1320, MIT and RSA Data Security, Inc., April.

Rochlis, J., and M. Eichin (1989) "With Microscope and Tweezers: The Worm from MIT's Perspective," *Communications of the ACM*, **32**(6):689–698, June.

Roman, Steven (1999) *Writing Word Macros*, Sebastopol, CA, O'Reilly Assoc.

Rosenberger (2005) is URL http://www.vmyths.com/fas/fas1.cfm.

RSA (2001) is URL http://www.rsasecurity.com/rsalabs/challenges/factoring/ file faq.html.

RSA-MD4 (2005) is http://www.rsasecurity.com/rsalabs/node.asp?id=2253.

RSASecurID (2004) is http://www.rsasecurity.com/node.asp?id=1157.

RSAsecurity (2004) is URL http://www.rsasecurity.com/.

safekids (2005) is URL http://www.safekids.com/.

SaftLite (2005) is available at http://haoli.dnsalias.com/shared/saft_lite.hqx.

Salomon, David (2003) *Data Privacy and Security*, New York, Springer Verlag.

Salomon, David (2004) *Data Compression: The Complete Reference*, 3rd edition, New York, Springer Verlag.

Salomon, David (2005) *Coding for Data and Computer Communications*, New York, Springer Verlag.

Savard (2005) is URL http://home.ecn.ab.ca/~jsavard/crypto/jscrypt.htm.

scc-inc (2004) is `http://www.scc-inc.com/SccVsLexmark/`.

Schneier, Bruce (1993) "Fast Software Encryption," in *Cambridge Security Workshop Proceedings*, pp. 191–204. New York, Springer-Verlag. Also available from `http://www.counterpane.com/bfsverlag.html`.

Schneier, Bruce (1995) *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd edition, New York, John Wiley.

Schneier, Bruce (2003) is URL `http://www.counterpane.com/crypto-gram.html`.

Schneier, Bruce (2004) *Secrets and Lies: Digital Security in a Networked World*, Hoboken, NJ, John Wiley & Sons.

Schnorr, Claus Peter (1991) "Efficient Signature Generation for Smart Cards," *Journal of Cryptology*, **4**(3)161–174.

Schotti, Gaspari (1665) *Schola Steganographica*, Jobus Hertz, printer. Some page photos from this old book are available at URL `http://www.cl.cam.ac.uk/~fapp2/steganography/steganographica/index.html`.

SearchAndDestroy (2004) is URL `http://www.SearchAndDestroy.com/`.

secunia (2004) is `http://secunia.com/`.

send-safe (2005) is URL `http://www.send-safe.com`.

Shamir, Adi (1979) "How to Share a Secret," *Communications of the ACM*, **22**(11)612–613, November.

Shamir, Adi and Eran Tromer (2004) "Acoustic cryptanalysis," available online in html format at URL `http://www.wisdom.weizmann.ac.il/~tromer/acoustic/`.

Shannon, Claude E. (1949) "Communication Theory of Secrecy Systems," *Bell System Technical Journal*, **28**:656–715, October.

Shannon, Claude E. (1951) "Prediction and Entropy of Printed English," *Bell System Technical Journal*, **30**:50–64, January.

Shoch, John and Jon Hupp (1982) "The Worm Programs—Early Experience With a Distributed Computation," *Communications of the ACM*, **25**(3)172–180. Reprinted in [Denning 90].

Simovits, Mikael J. (1996) *The DES, an Extensive Documentation and Evaluation*, Laguna Hills, Calif., Aegean Park Press, vol. **C-68**.

Singh, Simon (1999) *The Code Book*, New York, Doubleday.

Sinkov, A. (1980) *Elementary Cryptanalysis: A Mathematical Approach* (New Mathematical Library, No. 22), Washington, D.C., Mathematical Assn. of America.

smartcardalliance (2005) is `http://www.smartcardalliance.org/industry_info/index.cfm`.

Skoudis (2005) is `http://infosecuritymag.com/skoudis`

snapfiles (2004) is URL http://www.snapfiles.com/.

sophos (2005) is URL www.sophos.com/virusinfo/hoaxes.

Sorkin, Arthur (1984) "Lucifer, A Cryptographic Algorithm," *Cryptologia*, **8**(1):22–41, January. An addendum is in **8**(3)260–261.

spam (2004) is URL http://media.hormel.com/templates/knowledge/knowledge.asp?catitemid=14&id=94.

spam abuse (2004) is URL http://spam.abuse.net/others/sites.shtml.

SpamArrest (2004) is URL http://www.SpamArrest.com.

spambob (2004) is http://spambob.com/.

spamcop (2005) is URL http://www.spamcop.com/.

spamgourmet (2004) is http://www.spamgourmet.com/.

Spamhaus (2005) is URL http://www.spamhaus.org/.

spamhauslasso (2005) is http://www.spamhaus.org/statistics.lasso.

SpectorSoft (2004) is http://www.SpectorSoft.com/.

Spybot (2004) is http://www.spybot.info/en/index.html.

Spy Sweeper (2005) is URL http://www.webroot.com/.

spywareguide (2004) is URL http://www.spywareguide.com/.

spywareguide-country (2004) is URL http://www.spywareguide.com/articles/country_code_extensions_look_u_45.html.

spywareinfo (2004) is URL http://www.spywareinfo.com/.

ssa-gov (2004) is URL https://s044a90.ssa.gov/apps6/isss/bp-7004home.jsp.

ssa-form (2004) is http://www.ssa.gov/online/ssa-7004.pdf.

ssa-stat (2005) is URL https://s044a90.ssa.gov/apps6a/isss/main.html.

Stallings, William (1998) *Cryptography and Network Security: Principles and Practice*, Englewood Cliffs, N.J., Prentice-Hall.

Staniford, Stuart, Vern Paxson, and Nicholas Weaver (2002) "How to 0wn the Internet in Your Spare Time," *Proceedings of the 11th USENIX Security Symposium (Security '02)*. Available online at http://www.icir.org/vern/papers/cdc-usenix-sec02/index.html.

Steganosaurus (2004) is URL http://www.fourmilab.to/stego/.

Stego (2005) is URL http://www.stego.com/.

Stoll, Clifford (1988) "Stalking the Wily Hacker," *Communications of the ACM*, **31**(5) 484–497, May.

Stoll, Clifford (1990) *The Cuckoo's Egg*, Bodley Head.

Stoll, Clifford (2004) `http://www.ocf.berkeley.edu/~stoll/`.

storagereview (2000) is URL
`http://www.storagereview.com/guide2000/ref/hdd/index.html`.

Strunk, William Jr. (1918) *The Elements of Style*, Ithaca, NY, W. P. Humphrey, (also NY, Bartleby.com 1999).

sweetcocoa (2005) is URL `http://homepage.mac.com/sweetcocoa/lapcop/`.

Symantec (2004) is URL `http://www.symantec.com/index.htm`.

takedown (2004) is URL `http://www.takedown.com/`.

technet (2004) is `http://www.microsoft.com/technet/archive/community/columns/security/essays/10imlaws.mspx`.

theinquirer (2004) is UTL `http://www.theinquirer.net/?article=19159`.

Thomas, Steven A. (2000) *SSL and TLS Essentials: Securing the Web*, New York, John Wiley.

Thompson, Ken (1984) "Reflections on Trusting Trust," *Communications Of The ACM*, **27**(8)172–180.

TransUnion (2005) is URL `https://www.freecreditprofile.com/`.

Trithemius, Johannes (1606) *Steganographia*. Available (for private use only) from URL `http://www.esotericarchives.com/tritheim/stegano.htm`.

truste (2005) is URL `http://www.truste.org/`.

Unicode (2005) is URL `http://www.unicode.org`.

Unicode Standard (1996) *The Unicode Standard*, Version 2.0, Reading, Mass., Addison-Wesley.

van Eck, Wim (1985) "Electromagnetic Radiation from Video Display Units: An Eavesdropping Risk," *Computers and Security*, **4**:269–286.

Verisign (2004) is `http://www.verisign.com/products-services/security-services/pki/unified-authentication/`.

versiontracker (2005) is URL `http://www.versiontracker.org/`.

Virus bulletin (2005) *Virus Bulletin: The International Publication on Computer Virus Prevention, Recognition, and Removal*. Available online at
`http://www.virusbtn.com/magazine/`.

vmyths (2005) is URL `http://www.vmyths.com/`.

vote-smart (2005) is `http://www.vote-smart.org/pdf/vsdm2004/vsdm-2004.pdf`.

WatermarkingWorld (2005) is located at URL `http://www.watermarkingworld.org/`.

Wayner, Peter (1992) "Mimic Functions," *Cryptologia*, **XVI**(3)193–214, July.

Wayner, Peter (2002) *Disappearing Cryptography*, 2nd edition, London, Academic Press.

Webopedia (2004) is URL `http://www.webopedia.com/`.

webroot (2004) is URL `http://www.webroot.com/`.

Western digital (2004) is URL
`http://www.wdc.com/en/products/Products.asp?DriveID=35`.

Wild List (2004) is URL `http://www.wildlist.org/`.

Witten, Ian H. (1987) "Computer (In)security: Infiltrating Open Systems," *ABACUS*, **4**(4)7–25. Also available from `http://cryptome.org/compinsec.htm` and in [Denning 90].

Wright, Peter (1989) *Spycatcher: The Candid Autobiography of a Senior Intelligence Officer*, New York, Random House.

Wyatt, Allen (2004) *Cleaning Windows XP for Dummies*, New York, John Wiley.

Zalewski, Michal (2005) *Silence on the Wire: A Field Guide to Passive Reconnaissance and Indirect Attacks*, San Francisco, CA, No Starch Press.

Zimmermann, Philip (1995) *PGP Source Code and Internals*, Cambridge, Mass., MIT Press.

Zimmermann, Philip (2001) is `http://www.philzimmermann.com/`.

When you re-read a classic you do not see in the book more than you did before. You see more in you than there was before.

—Clifton Fadiman

Index

This long index reflects this author's belief that a detailed index is invaluable in a scientific/technical book. A special effort was made to include full names (first and middle names instead of initials) and dates of persons mentioned in the book. Any mistakes, inaccuracies, and omissions found in the index and reported to the author will be included in the errata list and corrected in any future editions of the book.

If you don't find it in the index, look very carefully through the entire catalogue.
—*Sears, Roebuck, and Co. Consumer's Guide*, 1897

Colophon

The idea for this book was proposed to this author in mid 2004 by Wayne Wheeler, the computer science editor of Springer Verlag. Most of the material was written in late 2004 and early 2005. It is based on the author's own experience with computer security issues, on topics discussed in many books on computer security, and on material found on the Internet about recent problems and attacks and how to fight them. The chapter on cryptography has improved and extended material from [Salomon 03]. The many inserts with quotations have been included to liven up the book and also to push the text up or down in order to improve the page breaks.

The book was designed by the author and was typeset by him in plain TEX (plus about 150 macros). The figures and diagrams were drawn in Adobe Illustrator. The following numbers convey an idea of the amount of work that went into the book:

- The book contains about 180,000 words, consisting of about 1,100,000 characters.
- The text is typeset mainly in font cmr10, but about 30 other fonts were used.
- The raw index file has about 1850 items.
- There are about 190 cross references in the book.

As the Preface promises, this is not a fact-free book.

"I'm afraid I haven't read any of your books. I believe you write books, don't you? I hope you won't mind that."
I was in process of picking out one of the several routine replies designed to bridge this not at all uncommon conversational opening—a phrase that at once generously accepts the speaker's candour in confessing the omission, while emphasizing the infinite unimportance of any such solicitude on that particular point—when need to make any reply at all was averted by a matter of much greater interest to both of us. This was entry into the room of Widmerpool and the Quiggin twins.

—Anthony Powell, *Hearing Secret Harmonies* (1975)

Index

This long index reflects this author's belief that a detailed index is invaluable in a scientific/technical book. A special effort was made to include full names (first and middle names instead of initials) and dates of persons mentioned in the book. Any mistakes, inaccuracies, and omissions found in the index and reported to the author will be included in the errata list and corrected in any future editions of the book.

If you don't find it in the index, look very carefully through the entire catalogue.
—*Sears, Roebuck, and Co. Consumer's Guide*, 1897