Preface to the First Edition

> It is easy to run a secure computer system. You merely have to disconnect all dial-up connections and permit only direct-wired terminals, put the machine and its terminals in a shielded room, and post a guard at the door.
>
> —F.T. GRAMPP AND R.H. MORRIS

> Of course, very few people want to use such a host...
>
> —THE WORLD

For better or for worse, most computer systems are not run that way today. Security is, in general, a trade-off with convenience, and most people are not willing to forgo the convenience of remote access via networks to their computers. Inevitably, they suffer from some loss of security. It is our purpose here to discuss how to minimize the extent of that loss.

The situation is even worse for computers hooked up to some sort of network. Networks are risky for at least three major reasons. First, and most obvious, more points now exist from which an attack can be launched. Someone who cannot get to your computer cannot attack it; by adding more connection mechanisms for legitimate users, you are also adding more vulnerabilities.

A second reason is that you have extended the physical perimeter of your computer system. In a simple computer, everything is within one box. The CPU can fetch authentication data from memory, secure in the knowledge that no enemy can tamper with it or spy on it. Traditional mechanisms—mode bits, memory protection, and the like—can safeguard critical areas. This is not the case in a network. Messages received may be of uncertain provenance; messages sent are often exposed to all other systems on the net. Clearly, more caution is needed.

The third reason is more subtle, and deals with an essential distinction between an ordinary dial-up modem and a network. Modems, in general, offer one service, typically the ability to log in. When you connect, you're greeted with a `login` or `Username` prompt; the ability to do other things, such as sending mail, is mediated through this single choke point. There may be vulnerabilities in the *login* service, but it is a single service, and a comparatively simple one. Networked computers, on the other hand, offer many services: *login*, file transfer, disk access, remote execution, phone book, system status, etc. Thus, more points are in need of protection—points that are more complex and more difficult to protect. A networked file system, for example,

cannot rely on a typed password for every transaction. Furthermore, many of these services were developed under the assumption that the extent of the network was comparatively limited. In an era of globe-spanning connectivity, that assumption has broken down, sometimes with severe consequences.

Networked computers have another peculiarity worth noting: they are generally not singular entities. That is, it is comparatively uncommon, in today's environment, to attach a computer to a network solely to talk to "strange" computers. Organizations own a number of computers, and these are connected to each other and to the outside world. This is both a bane and a blessing: a bane, because networked computers often need to trust their peers, and a blessing, because the network may be configurable so that only one computer needs to talk to the outside world. Such dedicated computers, often called "firewall gateways," are at the heart of our suggested security strategy.

Our purpose here is twofold. First, we wish to show that this strategy is useful. That is, a firewall, if properly deployed against the expected threats, will provide an organization with greatly increased security. Second, we wish to show that such gateways are necessary, and that there is a real threat to be dealt with.

Audience

This book is written primarily for the network administrator who must protect an organization from unhindered exposure to the Internet. The typical reader should have a background in system administration and networking. Some portions necessarily get intensely technical. A number of chapters are of more general interest.

> Readers with a casual interest can safely skip the tough stuff and still enjoy the rest of the book.

We also hope that system and network designers will read the book. Many of the problems we discuss are the direct result of a lack of security-conscious design. We hope that newer protocols and systems will be inherently more secure.

Our examples and discussion unabashedly relate to UNIX systems and programs. UNIX-style systems have historically been the leaders in exploiting and utilizing the Internet. They still tend to provide better performance and lower cost than various alternatives. Linux is a fine operating system, and its source code is freely available. You can see for yourself how things work, which can be quite useful in this business.

But we are not preaching UNIX here—pick the operating system you know best: you are less likely to make a rookie mistake with it. But the principles and philosophy apply to network gateways built on other operating systems, or even to a run-time system like MS-DOS.

Our focus is on the TCP/IP protocol suite, especially as used on the Internet. This is not because TCP/IP has more security problems than other protocol stacks—we doubt that very much—rather, it is a commentary on the success of TCP/IP. Fans of XNS, DECnet, SNA, netware, and others have to concede that TCP/IP has won the hearts and minds of the world by nearly any measure you can name. Most of these won't vanish—indeed, many are now carried over IP links, just

as ATM almost always carries IP. By far, it is the heterogeneous networking protocol of choice, not only on workstations, for which it is the native tongue, but on virtually all machines, ranging from desktop personal computers to the largest supercomputers.

Much of the advice we offer in this book is the result of our experiences with our companies' intranets and firewalls. Most of the lessons we have learned are applicable to any network with similar characteristics. We have read of serious attacks on computers attached to public X.25 data networks. Firewalls are useful there, too, although naturally they would differ in detail.

This is not a book on how to administer a system in a secure fashion, although we do make some suggestions along those lines. Numerous books on that topic already exist, such as [Farrow, 1991], [Garfinkel and Spafford, 1996], and [Curry, 1992]. Nor is this a cookbook to tell you how to administer various packaged firewall gateways. The technology is too new, and any such work would be obsolete before it was even published. Rather, it is a set of guidelines that, we hope, both defines the problem space and roughly sketches the boundaries of possible solution spaces. We also describe how we constructed our latest gateway, and why we made the decisions we did. Our design decisions are directly attributable to our experience in detecting and defending against attackers.

On occasion, we speak of "reports" that something has happened. We make apologies for the obscurity. Though we have made every effort to document our sources, some of our information comes from confidential discussions with other security administrators who do not want to be identified. Network security breaches can be very embarrassing, especially when they happen to organizations that should have known better.

Terminology

> You keep using that word. I do not think it means what you think it means.
>
> Inigo Montoya in *The Princess Bride*
> —WILLIAM GOLDMAN [GOLDMAN, 1998]

Before we proceed further, it is worthwhile making one comment on terminology. We have chosen to call the attackers "*hackers*." To some, this choice is insulting, a slur by the mass media on the good name of many thousands of creative programmers. That is quite true. Nevertheless, the language has changed. Bruce Sterling expressed it very well [Sterling, 1992, pages 55–56]:

> The term "hacking" is used routinely today by almost all law enforcement officials with any professional interest in computer fraud and abuse. American police describe almost any crime committed with, by, through, or against a computer as hacking.
>
> Most important, "hacker" is what computer intruders choose to call *themselves*. Nobody who hacks into systems willingly describes himself (rarely, herself) as a "computer intruder," "computer trespasser," "cracker," "wormer," "dark-side hacker," or "high-tech street gangster." Several other demeaning terms have been invented in the hope that the press and public will leave the original sense of the word alone. But few people actually use these terms.

Acknowledgments

There are many people who deserve our thanks for helping with this book. We thank in particular our reviewers: Donato Aliberti, Betty Archer, Robert Bonomi, Jay Borkenhagen, Brent Chapman, Lorette Ellane Petersen Archer Cheswick, Steve Crocker, Dan Doernberg, Mark Eckenwiler, Jim Ellis, Ray Kaplan, Jeff Kellem, Joseph Kelly, Brian Kernighan, Mark Laubach, Barbara T. Ling, Norma Loquendi, Barry Margolin, Jeff Mogul, Gene Nelson, Craig Partridge, Marcus Ranum, Peter Weinberger, Norman Wilson, and of course our editor, John Wait, whose name almost, but not quite, fits into our ordering. Acting on all of the comments we received was painful, but has made this a better book. Of course, we bear the blame for any errors, not these intrepid readers.