Preface

We hope you enjoy our book, *A Practical Guide to Trusted Computing*. This is the first book available that guides you through the maze that is the Trusted Platform Module (TPM) now shipping from all major PC vendors. It also enables you to actually use the TPM.

What This Book Is About

This book is about the increasingly important discipline of Trusted Computing. As the number of viruses, Trojans, and spyware has increased over the last several years, so has the need for a way to provide safety to users. Although a number of books have been written that discuss the philosophy of trusted computing, this is the first one that gets down into the nitty gritty of what solutions can be afforded by making use of the Trusted Platform Modules (TPMs) and discussing how to code them. This book covers the basic capabilities of the TPM and how to write code that accesses those capabilities using the standard TCG Software Stack. It also provides example problems and discusses solutions that could be coded using TPM capabilities.

During the writing of this book, several of the authors were also working on the extension of the TSS 1.1 specification to the TSS 1.2 specification. The latter provides access to new functionality afforded by the 1.2 TPM. This book covers the new capabilities in the 1.2 TPM in Chapter 14, "Administration of Trusted Devices," so that people who want to write code that will work on any TPM can avoid that chapter, and those who want to use the new functionality of the TPM 1.2 can use Chapter 14 along with the rest of the book.

The authors of this book are truly experts in the field, having either worked on the specifications, written TSS stacks for use by software, or written software that uses the TPM itself. Several have given seminars, taught classes, or written papers on the use of the TPM.

What You Need to Know Before Reading This Book

The code in the book is all based on the C language, so skill in reading C is a requirement for understanding any of the coding examples. Additionally, it is important that the reader have some understanding of cryptography—particularly the difference between symmetric and asymmetric keys, and cryptographic hashes. There is some discussion in the book about these concepts, but a detailed description of the algorithms used is not included. Bruce Schneier's *Applied Cryptography* is a good reference for those who wish to go deeper into that area. If the reader wants merely to find out what TCG is good for, Parts I and III of the book are recommended. If the reader has a particular project in mind, all sections of the book are likely to be helpful.

Who You Are and Why You Should Read This Book

This book does provide the specific details needed to write applications that take advantage of the TPM. If you are unfamiliar with Trusted Computing and want to write code that will use the capabilities of the TPM, all of this book will be valuable to you. If you want to learn about the reasoning behind the design choices in the TPM, Parts I and II of the book are the ones to concentrate on.

For a Software Engineer

The authors have tried to write a book that would include everything they would have liked to know about programming with TPMs. As a result, we have included sample code that we have compiled to make sure it works. We give examples that do real things, not just using defaults everywhere. We explain the choices we make in designing the code, and the code is commented, so it is clear about what it does.

If you want to understand how big the problem is that needs to be solved, read Chapter 1, "Introduction to Trusted Computing." If you want to learn about the capabilities of the TPM, read Chapter 2, "Design Goals of the Trusted Platform Module," and Chapter 3, "An Overview of the Trusted Platform Module Capabilities." If you want to find out what kinds of problems can be solved using the capabilities of the TPM, read Chapters 11–13. If you already understand the capabilities of the TPM and want to write programs that use TPM 1.1, read Chapters 4–10. If you want to use the expanded capabilities in the TPM 1.2, read Chapter 14.

For a Software Project Manager or Technical Leader

A software project manager needs to understand the capabilities of the TPM and also the architecture of projects he is leading. In any security program, it is particularly important that the architecture be established well before coding begins. Architectural design flaws lead quickly to security flaws.

This book should help you understand the issues necessary to design a secure program architecture that takes advantage of the TPM. Chapters 1, 2, 3, 11, 12, 13, and 14 should be particularly useful for the project manager.

For a Computer User Interface Designer

Ease of use and security have been at odds ever since the first lock was designed. At first glance, they seem adamantly opposed to one another. The designs in Chapters 11, 12, and 13 may provide information necessary for the computer user interface designer to improve the usability of the solution.

For Those Interested in Trusted Computing

If a person is considering using the TPM, Chapters 1–3 and 11–13 provide the best reading. They provide a good overview of the problems that trusted computing tries to solve and how they are (architecturally) solved.

For Experienced Users of TPMs

If you are a long-time user of TPMs and are interested in what more you can do with the functions used in the TPM, this book—particularly Chapters 11, 12, 13, and 14—may provide you with inspiration. Sometimes just seeing how other people approach a problem is sufficient to provide the solution to a grating problem.

How the Book Is Organized

This section provides you with an overall view of how the book is organized, including a brief summary of all the chapters.

Part I: Background Material

Part I provides an overview of Trusted Computing, including what was the impetus that caused its creation, what problems it was trying to solve, and a functional view of what is provided by a Trusted Platform Module.

- **Chapter 1, "Introduction to Trusted Computing"**

  Historically, hackers have focused on the network, then the server, and now the client. This chapter gives an overview of the security attacks that are focused on today's client and their severity, and then explains why a TPM is ideal for solving such problems. It also discusses privacy issues and gives recommendations to the programmer to avoid causing privacy problems.

- **Chapter 2, "Design Goals of the Trusted Platform Module"**

  When the security experts who generated the original TPM got together, they had a number of features they wanted to make sure were included in the specification. This chapter discusses what they were trying to accomplish. Having this broad view of what a TPM was designed for will help the reader with background information necessary to understand how the actual features that were implemented were intended to be used.

- **Chapter 3, "An Overview of the Trusted Platform Module Capabilities"**

  This chapter describes the actual features implemented in the TPM 1.1 design and how they work. This provides an architectural view of the design of the specification. After reading this chapter, the reader should have an idea of what types of problems the TPM will help solve. Additionally, it provides some discussion as to why certain features were omitted from the specification.

Part II: Programming Interfaces to TCG

Part II includes chapters for the programmer. It provides an in-depth view of the interfaces that are available in the software stack, with examples of how they are used. It starts out with the lowest level—talking to the device driver. Next, it looks at the boot sequence for a computer and how that can be enhanced with a TPM. This is followed with a section on the core services provided by the software stack, along with a brief discussion of talking directly to this interface, as when a remote application is using a TPM. Following this, the next few chapters are about using the TPM at the highest level: the application interface

- **Chapter 4, "Writing a TPM Device Driver"**

  This chapter provides the reader with the information necessary to write a device driver to communicate with the TPM. This is important to the person who wishes to use the TPM with an operating system other than those (Windows, Linux) that already have device drivers available.

- **Chapter 5, "Low-Level Software: Using BIOS and TDDL Directly"**

  This chapter provides the reader with the information necessary to talk directly to the chip in the absence of a TSS stack. This is important to a person writing code that runs in BIOS, or for writing a TSS stack for a new operating system, or in a memory constrained environment. This chapter is based on work originally done for Linux, but has been deliberately written in an OS neutral format. Additionally, this chapter will provide to the user a real appreciation of the work done for him when he is using the TSS stack.

- **Chapter 6, "Trusted Boot"**

  This chapter describes using the chip in one of the most exciting forms—to measure the security state of a platform. There are two means of doing this in the Trusted Computing space: the 1.1 Static Root of Trust and the 1.2 Dynamic Root of Trust. Both are described in detail, and example code is given showing how they are implemented. This is one of the few places in the book where 1.2 code that has been tested can be given, as these interfaces do not require nonexistent 1.2 TSS stacks.

- **Chapter 7, "The TCG Software Stack"**

  The TSS API is the most commonly used interface to access the TPM. This chapter describes the architecture of the TSS, conventions used in the API, and software object

types and their uses. It will also walk you through some simple programming examples using the TSS API and clarify the differences in programming for the 1.1 API versus the 1.2 API.

- **Chapter 8, "Using TPM Keys"**

  Key management is one of the most difficult things to achieve in a security program, and one of the areas where a TPM excels. This chapter describes and gives examples of how keys can be created, stored, loaded, migrated, and used. Special keys are described, including identity keys, storage keys, and signing keys, along with examples of their usage.

- **Chapter 9, "Using Symmetric Keys"**

  This chapter explains the richness of facilities provided by the TPM to exploit symmetric keys in applications. The reader who is interested in using the TPM to enhance the security of any application that does bulk encryption will want to read this chapter to find out, with examples, how to exploit TPM features.

- **Chapter 10, "The TSS Core Service (TCS)"**

  The core services underlie the normal application interface APIs. It is important that an application developer have some idea of what these services provide, so that he can understand exactly what is happening when an API is called. Additionally, if an application writer wants to create a client server application, and a TPM needs to be asked to perform services remotely, the core services are the application layer that is called. This chapter provides insight into the core services and some sample code for doing remote calls.

- **Chapter 11, "Public Key Cryptography Standard #11"**

  This chapter provides a real example of coding using the TSS. It provides a complete working example of a PKCS#11 stack that links to the TSS stack to provide middleware services to applications. The code is commented and is available open source for use.

Part III: Architectures

This section of the book is intended to give the reader a flavor of the richness of applications that are enabled with the Trusted Computing software stack. It will provide the reader with an idea of the target applications that the specification writers had in mind when they designed the architecture. Even if the reader is not interested in writing these particular applications, reading these chapters will help explain why design decisions were made the way they were.

- **Chapter 12, "Trusted Computing and Secure Storage"**

  The TPM provides capabilities for storing data securely in two commands: BIND and SEAL. This chapter provides a number of examples of how those commands could be used to provide useful functions to an end user. It also discusses some of the problems

that need to be solved to have a secure implementation. Reading this chapter will help the reader understand why the commands were designed the way they were.

- **Chapter 13, "Trusted Computing and Secure Identification"**

  The TPM provides capabilities for doing secure signing internal to the chip itself. This chapter gives a number of examples of how those functions can be used to provide practical applications that solve real user problems. Reading this chapter will help the reader understand why the signing commands were designed the way they were.

- **Chapter 14, "Administration of Trusted Devices"**

  When companies start deploying TPMs in large numbers, it becomes particularly important that they find ways of administrating them. This chapter is concerned with how to use the migration commands to provide administration applications that will allow remote administration of a TPM.

- **Chapter 15, "Ancillary Hardware"**

  The TPM could not, by itself, solve all security problems. It is by design an inexpensive device. However, it does provide a number of capabilities that allow it to hook into other security devices. This chapter describes some of the ways this can be done to provide enhanced security for a client.

- **Chapter 16, "Moving from TSS 1.1 to TSS 1.2"**

  The TSS 1.2 specification has recently been released. In this chapter, we describe the new capabilities that are in the new specification, and give code examples of how each one can be used. These new capabilities include CMK, Delegation, DAA, new PCR behavior, Locality, NVRAM, Audit, Monotonic counter, Transport, Tick, and Administrative commands This section is aimed at those who want to take advantage of these new features to write code that will work ONLY on clients that include 1.2 TPMs.

Part IV: Appendixes

We have also included several useful appendixes. If you are just trying to find out an API that will provide a specific function, these appendixes give a quick way of finding the APIs in question. For those looking to talk to the hardware directly, a TPM command reference is given. For those looking to talk to the Trusted Computing Software Stack, a TSS command reference is given. These references provide a quick description of the command and a list of how they are typically designed to be used.

- **Appendix A, "TPM Command Reference."**

  A list of TPM-level commands, where they were intended to be used, and a brief description of what they do.

- **Appendix B, "TSS Command Reference."**

  A list of TSS-level commands, where they were intended to be used, and a brief description of what they do.

- **Appendix C, "Function Library."**

  A list of proposed helper functions which would help the user in creating programs using the TPM, along with a description of their function.

- **Appendix D, "TSS Functions Grouped by Object and API Level."**

  This appendix breaks down the functions by the internal TSS object that they affect and by the API level at which they interact. This information can be used as a quick table for determining which APIs are available while writing code.

Some reviewers have noted that considering possible uses for the TPM helped them understand the reasoning behind the design of the TPM. The authors hope that this book will not only help in that understanding, but promote the exploitation of the resource that is now so widely deployed.