Preface

The value of information and the power that it can convey has long been recognised. Now, more than ever, information is a driver of society and often its integrity, confidentiality and authenticity must be ensured.

Security protocols are a critical element of the infrastructures needed for the secure communication and processing of information. They are of course not the only components needed to ensure such security properties: for example, good cryptographic algorithms and systems security measures to protect key material are also needed. Protocols can however be thought of as the keystones of a secure architecture: they allow agents who may be remote to authenticate each other, to establish fresh session keys to communicate confidentially, to ensure the authenticity of data and services and so on.

Aims of the book

This book is about the role of security protocols, how they work, the security properties they are designed to ensure and how to design and analyse them.

It was recognised very early on, almost as soon as they were conceived, that the design and analysis of security protocols was going to be a very delicate and error prone process. Security protocols are deceptively simple looking objects that harbour surprising subtleties and flaws. Attempts to develop frameworks and tools to reason about their properties goes back over 20 years but the topic remains a highly active and fruitful one in the security research community. An overview of the historical background can be found in Chapter 9.

In this book we present the particular approach to security protocol verification that has been developed by the authors. It was the first to apply process algebra and model-checking to the problem. The process algebra in question is CSP (Communicating Sequential Processes)

There is a widespread misconception that pouring liberal libations of cryptographic algorithms over an architecture will render it secure. Certainly good cryptographic algorithms are important but, as we will see, it is quite possible to have an architecture employing high grade algorithms that is still wide open to exploitation due to protocol design.

We hope that our readers will come away with a good understanding of the role of security protocols, how they work and the kinds of vulnerabilities to which they

are prey. In particular we hope that they will better appreciate the subtleties in making precise the security goals that such protocols are intended to ensure and the importance of making these goals as well as the assumptions about the underlying mechanisms and environment precise.

Ideally we hope that the reader will gain sufficient understanding (and enthusiasm!) to apply the tools and techniques presented here to their own protocols, real or imaginary. Perhaps also some readers will be sufficiently intrigued to go on to pursue research into some of the open problems that remain in this challenging and fascinating area.

Structure of the book

This book is concerned with the particular approach to analysis and verification of security protocols based around the process algebra CSP. There are a number of facets to this approach, and the book uses a running example, the Yahalom protocol, to link the material.

The Introduction (Chapter 0) introduces the general topic of security protocols. It covers the issues that arise in their design, the cryptographic mechanisms that are used in their construction, the properties that they are expected to have, and the kinds of attacks that can be mounted to subvert them. It also discusses the CSP approach and the tool support. The chapter introduces the Yahalom protocol and several other protocol examples.

Chapter 1 provides a general introduction to the main aspects of CSP relevant to the approach. CSP consists of a language and underlying theory for modelling systems consisting of interacting components, and for supporting a formal analysis of such models. This chapter introduces the building blocks of the language which enable individual components to be described, and discusses how components are combined into systems. Specification and verification through refinement, and with respect to property-oriented specifications, is also covered. The chapter finishes with a brief discussion of how discrete time can be modelled.

Chapter 2 shows how to use CSP to construct models of security protocols, which consist of a number of communicating components and thus well-suited to analysis in CSP. The variety of possible attacks on protocols must also be built into the model, and the chapter shows how to incorporate the Dolev-Yao approach to modelling a hostile environment, and produce a system description which is suitable for analysis.

Chapter 3 covers the kinds of properties that security protocols are expected to provide, and how they can be expressed formally within the CSP framework. Secrecy and authentication are the main concern of the approaches in this book, and various forms are covered. The properties of non-repudiation and anonymity are also discussed.

Chapter 4 introduces the model-checking tool support available for CSP, the Failures-Divergences Refinement checker (FDR). It discusses how this tool works, and the nature of refinement-checking.

Chapter 5 is concerned with the Casper tool. This is a compiler for security protocols, which transforms a high-level description of a security protocol and the properties required of it, into a CSP model of the protocol as described in Chapter 2, and a number of assertions to be checked. This model can then be analysed using the model-checker FDR discussed in Chapter 4.

Chapter 6 discusses in more detail some of the CSP modelling which is carried out by Casper, particularly how the hostile environment is modelled to allow efficient analysis by the model-checker.

Chapter 7 is concerned with direct verification of CSP models of protocols. It introduces the 'rank function' approach to proving protocols correct. This allows proofs to be constructed which verify protocol descriptions of arbitrary size against their requirements. The theorem-proving and bespoke tool support available for this approach is also discussed.

Chapter 8 addresses the problem of scale. Real-world protocols are very large and their analysis is difficult because of the volume of detail contained in their description. This chapter is concerned with 'simplifying transformations' which allow extraneous detail to be abstracted away when checking a protocol against a particular property in such a way that verification of the abstract protocol implies correctness of the full protocol. The approach is illustrated with the CyberCash main sequence protocol.

Chapter 9 discusses the literature on security protocol verification and its historical context. There are a number of different approaches to the problems addressed in this book, and this chapter covers many of those that have been most influential in the field.

Chapter 10 discusses the broader issues, open problems and areas of ongoing research, and gives indications of areas for possible further developments and research. One area of current research discussed in this chapter, of particular importance to the model-checking approach of this book, is the development of techniques based on *data independence*, which allow the results of model-checking to be lifted to protocol models of arbitrary size.

There are three appendices. The first covers some background mathematics and cryptography, introducing the RSA and the ElGamal systems for creating cryptographic keys; the second is an example of Casper applied to the Yahalom protocol, containing the input file and the CSP model produced by Casper; and the third contains a verification using rank functions of the simplified CyberCash protocol descriptions produced in Chapter 8.

The book has an associated website:

`http://www.cs.rhbnc.ac.uk/books/secprot/`

This website provides access to all of the tools discussed in this book, and to the protocol examples which are used throughout (as well as others). Readers are recommended to download the tools and experiment with protocol analysis while reading the book. The website also provides exercises (and answers!), as well as a variety of other related material.