

Preface

In the setting of multiparty computation, sets of two or more parties with private inputs wish to jointly compute some (predetermined) function of their inputs. The computation should be such that the outputs received by the parties are correctly distributed, and furthermore, that the privacy of each party's input is preserved as much as possible, even in the presence of adversarial behavior. This encompasses any distributed computing task and includes computations as simple as coin-tossing and broadcast, and as complex as electronic voting, electronic auctions, electronic cash schemes and anonymous transactions. The feasibility (and infeasibility) of multiparty computation has been extensively studied, resulting in a rather comprehensive understanding of what can and cannot be securely computed, and under what assumptions.

The theory of cryptography in general, and secure multiparty computation in particular, is rich and elegant. Indeed, the mere fact that it is possible to actually achieve the aforementioned task is both surprising and intriguing. However, the focus of this book is not on the theory of secure computation (although a number of results with theoretical importance are studied here), but rather on the question of *efficiency*. Recently, there has been increasing interest in the possibility of actually using secure multiparty computation to solve real-world problems. This poses an exciting challenge to the field of cryptography: Can we construct secure protocols (with rigorous proofs of security) that are truly efficient, and thus take the theory of secure computation to the next step towards practice. We stress that this book is not about “practical cryptography”. We do not take systems considerations into account, nor how protocols should be implemented and deployed. Instead, our aim is to provide an introduction to the field of efficient protocol construction and design. We hope that this book will make the field of secure computation in general, and efficient protocol construction in particular, more accessible and will increase awareness regarding the importance of this vibrant field.

Outline. This book is divided into three distinct parts:

- **Introduction and definitions:** We begin with a general introduction and survey of secure computation, followed by definitions of security under a number of different adversary models. This part also includes important material regarding the properties of these definitions, and the relations between them.
- **General constructions:** In this part, we present secure protocols for general secure computation. That is, we present protocols that can be applied to any circuit computing any efficient function. Although this does not enable us to utilize specific properties of the function being computed, the resulting protocols can be efficient enough if the circuit and input are not too large.
- **Specific constructions:** Finally, we study secure protocols for specific problems of interest. Two of the chapters in this part consider efficient constructions of basic building blocks that are widely used in constructions of secure protocols; namely, zero-knowledge (via Σ protocols) and oblivious transfer. The last two chapters study two specific examples of higher-level protocols; specifically, the secure computation of the k th ranked element (or median) of a distributed list, and secure search operations on databases. The constructions in this part demonstrate how specific properties of a function being computed can be utilized to achieve greater efficiency.

It goes without saying that the material presented in this book is far from an exhaustive study of results in the field. There are many alternative constructions achieving some of the results presented here, and many other problems of interest for which efficient protocols have been constructed. In some places throughout, we have added pointers to additional readings of relevance.

In order to not unnecessarily complicate the constructions and models, we have focused on the *two-party* case and consider only *static adversaries* and the *stand-alone model*. We do not claim that this is the best model for constructing protocols; indeed it is arguably too weak in many cases. However, we believe that it serves as a good setting for an initial study, as it is significantly cleaner than other more complex settings.

Prerequisite knowledge. We assume that the reader is familiar with the basics of theoretical cryptography. Thus, for example, we assume that readers know what commitment schemes and zero-knowledge proofs are, and that they are comfortable with notions like pseudorandomness and computational indistinguishability. In contrast, all the relevant definitions of secure two-party computation are presented here from scratch. Thus, this book can also be used as a first introduction to secure computation.

Reading this book. Although there are advantages to reading this book in sequential order, much of the book can be read “out of order”. It goes without saying that the chapter on definitions is needed for all later chapters. However, it is possible to read definitions as needed (e.g., read Section 2.2

and then Chapter 3, then Section 2.3 followed by Chapter 4, and so on). Regarding the general constructions in Part II of the book, the constructions in Chapters 4 and 5 rely in a direct way on Chapter 3, and thus it is highly recommended to read Chapter 3 first. In contrast, Chapters 4 and 5 can be read independently of each other.

The specific constructions in Part III can be read independently of the general constructions in Part II. It is preferable to read Chapters 6 and 7 first (and in order) because later protocols use the tools introduced in these chapters. In addition, some of the oblivious transfer protocols of Chapter 7 use zero-knowledge proofs that are constructed in Chapter 6. Nevertheless, if one is satisfied with referring to an arbitrary zero-knowledge proof or oblivious transfer protocol, then the chapters in Part III can be read in any order.

Book aims and its use for teaching a course. This book can be used as a textbook for an introductory course on secure computation with a focus on techniques for achieving efficiency, as an entry point for researchers in cryptography and other fields like privacy-preserving data mining who are interested in efficient protocols for secure computation, and as a reference for researchers already in the field. Regarding its use as a textbook, due to the flexibility regarding the order of reading this book (as described above), it is possible to design courses with different focuses. For example, a more theoretical course would spend considerable time on definitions and the general constructions of Part II of the book, whereas a more applied course would focus more on the specific constructions in Part III. We remark also that Chapters 6 and 7 can serve as a nice opening to a course; the material is not as heavy as general secure computation and contains many interesting ideas that can be attractive to students. When teaching a general introduction to (computational) secure computation, it is certainly possible to base much of the course on this book. However, in such a case we would also teach the GMW construction. A full treatment of this appears in [35, Chapter 7].

Comments and errata. We will be more than happy to receive any (positive or negative) feedback that you have on this book, as well as any errors that you may find. Please email us your comments and errata to lindell@cs.biu.ac.il. A list of known errata will be maintained at <http://www.cs.biu.ac.il/~lindell/efficient-protocols.html>.

Acknowledgements. First and foremost, we would like to thank Ivan Damgård for generously providing us with the text that formed the basis of Chapter 6 on Σ protocols. In addition, we would like to thank Oded Goldreich, Jonathan Katz and Eran Omri for providing us with constructive advice and comments on this book.

Carmit Hazay: First, I would like to thank my co-author Yehuda Lindell who was also my Ph.D. advisor. Yehuda introduced me to the area of secure computation and has greatly contributed to my academic career. He is a continuing source of inspiration and assistance, and I am grateful to him for an amazing journey which led to this book.

During my Ph.D. I had the pleasure of working with many talented people who enriched my knowledge and deepened my understanding regarding secure computation. I would like to thank Ran Canetti, Rosario Gennaro, Jonathan Katz, Hugo Krawczyk, Kobbi Nissim, Tal Rabin and Hila Zarosim for many productive discussions and a memorable time.

Yehuda Lindell: First and foremost I would like to thank Oded Goldreich. Beyond being my Ph.D. advisor, and as such of great influence on my academic career, Oded has continued to provide valuable support, advice and encouragement. I owe much to Oded and am greatly indebted to him.

The ability to write this book is due to the knowledge that I have gained over many years of research in the field of secure computation. In this time, I have worked with many different co-authors and have benefited from countless fruitful discussions with many members of our research community. I would like to thank Yonatan Aumann, Boaz Barak, Ran Canetti, Rosario Gennaro, Shafi Goldwasser, Shai Halevi, Carmit Hazay, Yuval Ishai, Yael Kalai, Jonathan Katz, Eyal Kushilevitz, Hugo Krawczyk, Tal Malkin, Moni Naor, Benny Pinkas, Tal Rabin, Alon Rosen and Adam Smith for years of joint work and cooperation in a friendly and enjoyable environment. Finally, I would like to give a special thanks to Benny Pinkas for all I have learned from him regarding topics of efficiency in secure protocols.

My work on this project was supported by the Israel Science Foundation (grant 781/07) and by a starting grant from the European Research Council.

October 2010

Carmit Hazay and Yehuda Lindell



<http://www.springer.com/978-3-642-14302-1>

Efficient Secure Two-Party Protocols
Techniques and Constructions

Hazay, C.; Lindell, Y.

2010, XIII, 263 p., Hardcover

ISBN: 978-3-642-14302-1