

SPRINGER BRIEFS IN COMPUTER SCIENCE

Máté Horváth · Levente Buttyán

Cryptographic Obfuscation

A Survey



Springer

SpringerBriefs in Computer Science

Series Editors

Stan Zdonik, Brown University, Providence, RI, USA

Shashi Shekhar, University of Minnesota, Minneapolis, MN, USA

Xindong Wu, University of Vermont, Burlington, VT, USA

Lakhmi C. Jain, University of South Australia, Adelaide, SA, Australia

David Padua, University of Illinois Urbana-Champaign, Urbana, IL, USA

Xuemin Sherman Shen, University of Waterloo, Waterloo, ON, Canada

Borko Furht, Florida Atlantic University, Boca Raton, FL, USA

V. S. Subrahmanian, University of Maryland, College Park, MD, USA

Martial Hebert, Carnegie Mellon University, Pittsburgh, PA, USA

Katsushi Ikeuchi, University of Tokyo, Tokyo, Japan

Bruno Siciliano, Università di Napoli Federico II, Napoli, Italy

Sushil Jajodia, George Mason University, Fairfax, VA, USA

Newton Lee, Institute for Education, Research and Scholarships, Los Angeles, CA, USA

SpringerBriefs present concise summaries of cutting-edge research and practical applications across a wide spectrum of fields. Featuring compact volumes of 50 to 125 pages, the series covers a range of content from professional to academic.

Typical topics might include:

- A bridge between new research results, as published in journal articles, and a contextual literature review
- A snapshot of a hot or emerging topic
- An in-depth case study or clinical example
- A presentation of core concepts that students must understand in order to make independent contributions

Briefs allow authors to present their ideas and readers to absorb them with minimal time investment. Briefs will be published as part of Springer's eBook collection, with millions of users worldwide. In addition, Briefs will be available for individual print and electronic purchase. Briefs are characterized by fast, global electronic dissemination, standard publishing contracts, easy-to-use manuscript preparation and formatting guidelines, and expedited production schedules. We aim for publication 8–12 weeks after acceptance. Both solicited and unsolicited manuscripts are considered for publication in this series.

****Indexing: This series is indexed in Scopus and zbMATH ****

More information about this series at <http://www.springer.com/series/10028>

Máté Horváth · Levente Buttyán

Cryptographic Obfuscation

A Survey

 Springer

Máté Horváth
Department of Networked
Systems and Services
Budapest University of Technology
and Economics (BME-HIT)
Budapest, Hungary

Levente Buttyán
Department of Networked
Systems and Services
Budapest University of Technology
and Economics (BME-HIT)
Budapest, Hungary

ISSN 2191-5768 ISSN 2191-5776 (electronic)
SpringerBriefs in Computer Science
ISBN 978-3-319-98040-9 ISBN 978-3-319-98041-6 (eBook)
<https://doi.org/10.1007/978-3-319-98041-6>

© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

To our teachers

Preface

“The Lord searches every heart and understands every desire and every thought.”

1 Chronicles 28:9, NIV

The ambitious goal of cryptographic obfuscation is to hide the operation of computer programs. Being an applied science, problems considered by cryptography are rarely investigated from a philosophical point of view but in the case of obfuscation, probably it worth spending some time considering the consequences of achieving this goal. The possibility of securely obfuscating arbitrary functions could radically change the relationship between humans and computer programs. Namely, it would imply losing our insight into the programs which we have had, at least in principle, since the writing of the first program code. While this change still seems to be futuristic, recent cryptographic advancements made it more probable than ever before.

In 2013 the breakthrough result of Garg, Gentry, Halevi, Raykova, Sahai and Waters (FOCS 2013) changed the previously pessimistic attitude towards general-purpose cryptographic obfuscation. Their finding was twofold. First, they managed to construct an obfuscator candidate that works for any function, which nonetheless was based on a rather idealistic assumption, and they showed a way to address the problem that had seemed impossible earlier. But what was probably even more important, they also demonstrated that their new tool is indeed useful and can help to solve other cryptographic problems as well. This latter observation was especially surprising as the security guarantee they achieved (called indistinguishability obfuscation) did not seem to have a practical relevance previously. An avalanche began and obfuscation became a central hub of cryptographic research. Cryptology ePrint Archive, the most active manuscript sharing forum of the community, counted over 190 related papers four years after the breakthrough, while before that fewer than 30 dealt with the topic. The potential realizability of such a powerful tool motivated a

plethora of applications, including solutions for long-standing open problems, from almost all areas of cryptography. At the same time, intense development of candidate constructions started with the double goal of basing the security of obfuscation on solid foundations and turning its incredible overhead into tolerable.

While these goals were still not achieved when finalizing our manuscript, the “obfuscation-fever” has already led us much closer to the root of hardness behind encrypted computations. However, looking up and understanding the key thoughts from an already huge number of articles that themselves are looking for the right definitions, methods, and formulations can be really troublesome and time-consuming. This challenge, which we also had to face, motivated us to review the rapid development of candidate obfuscator constructions and organize the results of the first years since the breakthrough. As the field is still changing rapidly, our work is not intended to be a retrospection but rather a handrail for those who are fascinated by the incredible opportunities offered by obfuscation and would like to catch up with the latest results by understanding their background.

We hope that our survey can reflect the beauty of the field and the reader will find answers for many of his or her questions in it.

Budapest,
November 2018

Máté Horváth
Levente Buttyán

Acknowledgements

First of all, we would like to thank our families for their patience. In this regard, special thanks goes to Judit. We are grateful to Ágnes Kiss, Örs Rebák and members of the CrySys Lab for their efforts to help us improve this work. We appreciate the valuable questions and remarks of Ryo Nishimaki, Ran Canetti, Zvika Brakerski and unknown reviewers that either highlighted flaws in earlier versions of our manuscript or helped us to better understand certain problems. Finally, we would also like to acknowledge the support of the National Research, Development and Innovation Office – NKFIH of Hungary under grant contract no. 116675 (K).

Contents

| | |
|---------------------------------------------------------------------------------|----|
| Glossary | xv |
| 1 Introduction | 1 |
| 1.1 Goals and Challenges | 1 |
| 1.2 Related Concepts – A Brief Comparison | 3 |
| 1.3 The Cryptographic Approach | 5 |
| 1.4 Milestones in Cryptographic Obfuscation | 7 |
| 1.5 This Survey and Related Literature | 9 |
| 1.5.1 Organization | 9 |
| 1.5.2 Related Work | 9 |
| 1.5.3 On the Used Notation | 9 |
| 2 Background | 11 |
| 2.1 Representation of Programs | 11 |
| 2.1.1 The Circuit Model of Computation | 11 |
| 2.1.2 Matrix Branching Programs | 12 |
| 2.2 The Cryptographic Primitives Used | 14 |
| 2.2.1 Fully Homomorphic Encryption | 14 |
| 2.2.2 Functional Encryption | 14 |
| 2.2.3 Randomized Encodings | 16 |
| 2.2.4 Multilinear Maps and Graded Encodings | 17 |
| 2.2.5 Simple and Efficient Pseudo-Random Generators | 20 |
| 2.2.6 Puncturable Pseudo-Random Functions | 21 |
| 2.3 Behind the Scenes of Security Proofs: Assumptions and Security Models | 22 |
| 2.3.1 On the “Desirable” and Actual Assumptions behind Obfuscation | 22 |
| 2.3.2 The Idea of Ideal Models | 24 |
| 2.3.3 Idealizations vs Reality: Criticism and Interpretations | 25 |
| 2.3.4 Variants of Ideal GES Models | 26 |

- 3 Definitional Approaches** 29
 - 3.1 Security via Simulation 29
 - 3.1.1 Virtual Black-Box Obfuscation 29
 - 3.1.2 Variants of the VBB Paradigm 30
 - 3.1.3 Evidence of VBB Impossibility 31
 - 3.1.4 Virtual Grey-Box Obfuscation 32
 - 3.2 Indistinguishability-Based Security 32
 - 3.2.1 Indistinguishability Obfuscation 32
 - 3.2.2 Different Faces of iO 33
 - 3.2.3 Relaxing the Efficiency Requirement: XiO 34
 - 3.2.4 Differing-Input or Extractability Obfuscation 35

- 4 Bootstrapping: From the Seed to the Flower** 37
 - 4.1 Amplifying Obfuscation with the Help of FHE 38
 - 4.1.1 Bootstrapping VBB Obfuscation 38
 - 4.1.2 From VBB to iO Bootstrapping 41
 - 4.2 Bootstrapping Obfuscation via Randomized Encodings 41
 - 4.2.1 The VBB Paradigm 42
 - 4.2.2 The Problem of Indistinguishably Obfuscating Probabilistic Circuits 42
 - 4.2.3 Full-Fledged iO from iO for Constant-Sized Circuits 43
 - 4.3 iO from Functional Encryption: An Alternative Pathway 44
 - 4.3.1 From FE to iO through Token-Based Obfuscation 45
 - 4.3.2 Multi-Input FE as an Intermediate Step 46
 - 4.3.3 A Classic Approach Using Compact RE 48
 - 4.4 Towards the Desired Compact FE 49
 - 4.4.1 iO-Based Bootstrappable FE 49
 - 4.4.2 From Secret-Key FE to Bootstrappable FE 50
 - 4.4.3 Compactness, Collusion Resistance, and the Role of PRGs 52

- 5 Building Core-Obfuscators – In Search of a Seed I.** 55
 - 5.1 Branching Program Obfuscation 56
 - 5.1.1 The Breakthrough Candidate iO Obfuscator 56
 - 5.1.2 Variants Secure in Pre-zeroizing Ideal Models 60
 - 5.1.3 Core-Obfuscators in the Standard Model 62
 - 5.2 Improving Efficiency: From MBP to Circuit Obfuscation 63
 - 5.2.1 Improving Efficiency by Minimizing MBP Size 64
 - 5.2.2 Direct Obfuscation of Circuits 65
 - 5.2.3 Implementing Obfuscation 66
 - 5.3 The Impact of GES Vulnerabilities on Core-Obfuscators 67
 - 5.3.1 Current Attacking Strategies 67
 - 5.3.2 Countermeasures 68

- 6 Building Functional Encryption: In Search of a Seed, II** 71
 - 6.1 Collusion-Resistant FE from the GGHZ Assumption 72
 - 6.2 iO from Constant-Degree GESs 73
 - 6.2.1 Circuit Obfuscation with a Constant Number of Multiplications 73
 - 6.2.2 Further Refinements 74
 - 6.3 FE for Low-Degree Polynomials from SXDH 75
 - 6.3.1 Computing Randomized Encodings with the Help of Inner Products 75
 - 6.3.2 Degree-Preserving FE 76
 - 6.4 Realization of PAFE 76

- 7 iO Combiners and Universal Constructions** 79
 - 7.1 Combiners for Obfuscation 79
 - 7.2 Universal iO 81

- References** 83

Glossary

| | |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| annihilating polynomial | A polynomial ρ is called the annihilating polynomial of a matrix A if $\rho(A) = 0$. |
| black-box technique | When constructing (or separating, i.e. proving the impossibility of a reduction) one cryptographic primitive \mathcal{P} from another one \mathcal{Q} , and we treat both \mathcal{Q} and the adversary \mathcal{A} as a black box (i.e. their code is not used), we say that the reduction from \mathcal{P} to \mathcal{Q} (or their separation) is black-box. Based on the extent of non-black-box techniques, several other notions of reducibility were defined by [RTV04] and refined by [BBF13]. |
| branching program | A branching program (BP) (a.k.a. binary decision diagram) is a DAG consisting of inner nodes of fan-out 2 labelled by Boolean variables l_i , including the source node (fan-in 0) and sinks of fan-out 0, labelled 0 or 1. The computation starts at the source and, at each node l_i , one proceeds to the other edge with label 0 if the i th input bit $x_i = 0$ or to the other if $x_i = 1$. The BP computes f if, for an input x , it reaches a sink, labelled by $f(x)$. A BP is <i>layered</i> if the nodes are partitioned into layers where the source is in the first layer and the sinks are in the last, and edges go only between nodes in consecutive layers. A permutation BP is a layered BP where all the nodes of a layer observe the same variable and the edges between any pair of consecutive layers form a permutation of the vertices (for any setting of the variables). See [Mit15, §5.8.1] and [Weg00]. |