SpringerBriefs in Computer Science

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

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

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 ith 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 *layered* 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].

coAM	The complexity class **coAM** is the complement of **AM**, which is the set of decision problems which are decidable in polynomial-time by a so-called Arthur–Merlin protocol (a specific interactive proof system) with two messages. See [AKG17].
CRS model	In the common reference string (CRS) model, it is assumed that everyone has access to a public string that is drawn from a predetermined distribution during a set-up phase.
factoring	The standard assumption of the hardness of factoring [Rab79] states that given $N = p_1 \cdots p_q$, where all p_i are random prime numbers of a given size, it is hard to find K such that $\gcd(K,N) \notin \{1,N\}$.
knowledge assumption	"Knowledge or extractability assumptions capture our belief that certain computational tasks can be done efficiently only by going through certain specific intermediate stages and generating some specific kinds of intermediate values. /.../ Though these assumptions do not fall in the class of falsifiable assumptions [Nao03], these have been proven secure against generic algorithms, thus offering some evidence of validity." [GS14, §8 (full version)]
learning with errors	The search/decisional learning with errors (LWE) assumption of [Reg05] states that it is hard to recover/distinguish a secret random vector $x \in \mathbb{Z}_p^n$ given noisy linear equations on it, i.e. given $y \in \mathbb{Z}_p^n$ and random $A \in \mathbb{Z}_p^{n\times m}$ such that $y = Ax + e \mod p$, where e is a random error vector of small magnitude. For its attractive features (e.g. suspected resistance to quantum attacks) and its connections to other assumptions, see [Pei16].
$\mathbf{NC^0}$	The class functions (also called local functions) which are computable by constant-depth, bounded-fan-in circuits, meaning that each output bit can only depend on a constant number of input bits. See [AKG17].
$\mathbf{NC^1}$	The class of polynomial-size circuits with logarithmic depth and bounded fan-in gates (more generally $\mathbf{NC^k}$ denotes the class of polynomial-size circuits of bounded fan-in having depth $O(\log^k n)$, where n is the input length). See [AKG17].
negligible function	$\mathsf{neg}(n)$ is called negligible if it grows more slowly than any polynomial, i.e. $\forall c \in \mathbb{N}, \exists n_0 \in \mathbb{N}$ such that $\forall n \geq n_0 :$ $\mathsf{neg}(n) < n^{-c}$.

NP	"**NP** is the class of decision problems solvable by a non-deterministic polynomial-time TM such that if the answer is 'yes,' at least one computation path accepts, but if the answer is 'no,' all computation paths reject" [AKG17].
NTRU	This is a public-key cryptosystem proposed by [HPS98] that is a possible alternative to factorization and discrete-log-based encryption schemes because of its efficiency and the fact that it is not known to be vulnerable to quantum attacks. [SS11] made it provably secure, assuming the hardness of worst-case problems over ideal lattices. The abbreviation refers to an *N*th-degree *tru*ncated polynomial ring, the underlying algebraic structure on which the cryptosystem is built.
one-way function	Informally speaking, a one-way function is a function that is easy to evaluate but hard to invert (on average). For further background, see [Gol06, §2].
one-way permutation	A one-way function that is a permutation (it is injective).
P	The class of decision problems solvable in polynomial-time by a Turing machine. See [AKG17].
P/poly	The class of polynomial-size circuits with unbounded depth (or, equivalently, polynomial-time TMs that take advice of polynomial length). See [AKG17] and [Gol08, §3.1].
proof system	A proof system consists of a *prover* and a *verifier*, where the prover aims to convince the verifier of a true statement. It is called "non-interactive" if the whole interaction between the parties is one message from the prover to the verifier. For details of the specific non-interactive witness-indistinguishable proofs used in the bootstrapping of obfuscation, see [FS90] and [GGH$^+$13b, §B.4]; for proof systems in general, see [Gol06, §4.10].
random oracle model	In this model, the cryptographic hash function is replaced by its ideal functionality: a truly random function, called a random oracle.
SAT	The Boolean satisfiability problem, which asks if there exists an assignment of variables in a given Boolean formula such that it evaluates to 1.
signature scheme	A signature scheme consists of three efficient algorithms: KeyGen (which outputs a signing and a verification key, *sk* and *vk*, respectively), Sign (which prepares a signature *s* for a message *m*, using *sk*), and verification (which on input (m,s) and *vk* outputs 1 if *s* is a valid signature of *m* under *sk*, and rejects otherwise). For the definition of its security, see the summary in [Gol06, §B.2].

SNARG	Succinct non-interactive arguments (SNARG) is a computationally sound (i.e. it is computationally infeasible to prove an assertion that is not true) proof system with short proofs for an NP-language. See [DSB17].
SNARK	Succinct non-interactive argument of knowledge (SNARK) is a SNARG system with the additional property that the correctness of a SNARK proof guarantees that the prover "knows" a witness to the statement with overwhelming probability. For details, see [BCC+17, DSB17].
standard model	In the standard, or plain, model, we assume that the adversary is limited only by the available amount of time and computational power.
$\mathbf{TC}^0$	$\mathbf{TC}^0 \subseteq \mathbf{NC}^1$ is the class of all Boolean circuits with constant depth and polynomial size, containing only unbounded-fan-in AND gates, OR gates, NOT gates, and threshold gates. See [AKG17].
trapdoor permutation	Intuitively, this is a one-way permutation with the extra property that, given some auxiliary information (the trapdoor), it is efficiently invertible. See [Gol06, §2.4.4].
Turing machine	The model of Turing machines captures all computational tasks that can be solved by classical computers. For details, see e.g. [Gol08, §1.2.3.2].

Acronyms

AS	Ananth–Sahai assumption
BGKPS	ideal graded encoding scheme (GES) model proposed by [BGK+14] (see Table 2.4)
BP	branching program
BPO	best-possible obfuscation
BR	ideal GES model proposed by [BR13] (see Table 2.4)
BSH	bounded speedup hypothesis
BSH′	parametrized bounded speedup hypothesis
CCA	chosen ciphertext attack model
CDH	computational Diffie–Hellman problem
CLT13	candidate GES type based on [CLT13]
CPA	chosen plaintext attack model
CRS	common reference string (see Glossary)
CRT	Chinese remainder theorem
d-MBP	dual-input matrix branching program (MBP)
DAG	directed acyclic graph
DDH	decisional Diffie–Hellman problem
DES	data encryption standard
DiO	differing-input obfuscation
Dlog	discrete logarithm problem
dRE	decomposable randomized encoding
EPI	equivalent program indistinguishability
ETH	exponential time hypothesis
(P_1, P_2, P_3, P_4)-FE	functional encryption with the properties defined in §2.2.2
FE	functional encryption
FHE	fully homomorphic encryption
$\mathcal{F}_{\mathrm{Lin}}$	function class defined by [Lin16] (see §4.4.1)
gcd	greatest common divisor
GCMM	generic coloured matrix model of [GGH+13b]

GES	graded encoding scheme
GGH13	candidate GES type based on [GGH13a]
GGH15	candidate GES type based on [GGH15]
GGHZ	the assumption proposed by [GGHZ16]
GGM	generic group model
gMBP	generalized MBP of [BMSZ16]
GMM+	"weak" ideal GES model proposed by [GMM$^+$16] (see Table 2.4)
IBE	identity-based encryption
iO	indistinguishability obfuscation
IPFE	inner-product functional encryption
jSXDH	joint SXDH
LWE	learning with errors (see the Glossary)
MBP	matrix branching program
MIFE	multi-input functional encryption
ML	machine learning
MMap	multilinear map
MPC	secure multi-party computation
MSE	multilinear subgroup elimination assumption
MSW-1	"multiplication restricted" ideal GES model of [MSW15] (see Table 2.4)
MSW-2	"non-restricted" ideal GES model of [MSW15] (see Table 2.4)
MSZ	"weak" ideal GES model proposed by [MSZ16] (see Table 2.4)
NIWI	non-interactive witness-indistinguishable proofs
NMiO	neighbouring-matrix iO
OWF	one-way function (see the Glossary)
PAFE	projective arithmetic functional encryption
pdRE	program-decomposable randomized encoding
PiO	probabilistic indistinguishability obfuscation (iO)
pk-FE	public-key functional encryption
PKE	public-key encryption
PPRF	puncturable pseudo-random function
PPT	probabilistic polynomial time
PRF	pseudo-random function
PRG	pseudo-random generator
PRG$^{X=z}$	polynomial-stretch pseudo-random generator (PRG) with complexity z according to the complexity measure X (see §2.2.5)
RAM	random access machine
RE	randomized encoding
rMBP	relaxed MBP of [AGIS14]
ROM	random oracle model (see the Glossary)
SD	subgroup decision assumption

SE	slotted encoding
SHE	somewhat homomorphic encryption
SiO	strong iO
sk-FE	secret-key functional encryption
SNARG	succinct non-interactive argument (see the Glossary)
SNARK	succinct non-interactive argument of knowledge (see the Glossary)
SSGES	semantic security of GESs
SSGES′	sub-exponential semantic security of GESs
SXDH	symmetric external Diffie–Hellman assumption
SXiO	strong exponentially efficient iO (XiO)
SXiO′	strong XiO with compression factor only slightly smaller than 1
TM	Turing machine (Glossary)
UC	universal circuit
VBB	virtual black-box
VGB	virtual grey-box
WBC	white-box cryptography
XiO	exponentially efficient iO