SpringerBriefs in Computer Science

Denise Demirel • Lucas Schabhüser
Johannes Buchmann

Privately and Publicly Verifiable Computing Techniques

A Survey

Springer

Denise Demirel
Theoretische Informatik
Technische Universität Darmstadt
Darmstadt, Hessen, Germany

Lucas Schabhüser
Theoretische Informatik
Technische Universität Darmstadt
Darmstadt, Hessen, Germany

Johannes Buchmann
Theoretische Informatik
Technische Universität Darmstadt
Darmstadt, Hessen, Germany

ISSN 2191-5768 ISSN 2191-5776 (electronic)
SpringerBriefs in Computer Science
ISBN 978-3-319-53797-9 ISBN 978-3-319-53798-6 (eBook)
DOI 10.1007/978-3-319-53798-6

Library of Congress Control Number: 2017931973

Printed on acid-free paper

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

Preface

Verifiable computing refers to methods that allow delegating the computation of a function on outsourced data to a server, such that the data owner and/or third parties can verify that the result has been computed correctly. Those approaches are even more useful when they provide a verification process that is more efficient than performing the computation locally. To address this challenge, many techniques have been proposed. In this work, the first comprehensive survey of all existing constructions is provided. In doing so, we are concerned with a setting where three parties are involved: A *client* who provides some input data, a *server* who evaluates a function on the input data, and a *verifier* who verifies the correctness of the result. Schemes dealing with a more complicated setting of multiple clients, verifiers, or servers are beyond the scope of this work. Furthermore, we do not consider approaches that rely on replication, trusted hardware, remote attestation, or spot checking.

For all approaches that match our setting and allow for a sufficiently efficient verification process, we provide a brief description of the approach and highlight the properties the solution achieve. More precisely, we analyze which level of security it provides, how efficient the verification process is, whether anyone or only the client can check the correctness of the result, which function class the verifiable computing scheme supports, and whether privacy with respect to the input and/or output data is given. Based on this analysis we compare the different approaches and outline possible directions for future work.

Darmstadt, Germany
June 2016

Denise Demirel
Lucas Schabhüser
Johannes Buchmann

Acknowledgments

This work has been co-funded by the European Union's Horizon 2020 research and innovation program under Grant Agreement No 644962. In addition, it has received funding from the DFG as part of project "Long-Term Secure Archiving" within the CRC 1119 CROSSING. The authors wish to thank Daniel Slamanig and David Derler for their valuable input and helpful discussion during the writing of this work.

Contents