

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



<http://www.springer.com/978-3-319-53797-9>

Privately and Publicly Verifiable Computing Techniques

A Survey

Demirel, D.; Schabhüser, L.; Buchmann, J.

2017, XII, 64 p., Softcover

ISBN: 978-3-319-53797-9