Journal of Computer Security 22 (2014) 659–660
DOI 10.3233/JCS-140505
IOS Press

Preface

Message from the Guest Editors

The theory of computer security is nowadays widely recognized as central to the challenge of making computers secure. Despite the progress in the last two decades, we are still lacking an overall understanding of the various aspects of computer security. Moreover, even for those areas for which we have established a solid theoretical understanding, we are often still lacking the technology to enforce the desired properties. This in particular affects the formal modeling as well as the verification of security protocols, the rigorous treatment of different attack class, as well as well-founded relations to other disciplines such as business objectives.

This special issue of the *Journal of Computer Security* includes four papers that cover various aspects of theoretical computer security, in particular the ones outlined above. Based on a full peer review process, these submissions were judged ready to be accepted to this special issue.

In "Obstruction-free authorization enforcement: Aligning security and business objectives", David Basin, Samuel J. Burri and Günter Karjoth analyzed the trade-off and the obstacles of using access control for business objectives. Amongst other contributions, they presented a novel approach to scoping authorization constraints within workflows with loops and conditional execution, formalized workflows, authorization constraints, and their enforcement using the process algebra CSP, visualized these constraints by extending the workflow modeling language BPMN, and provided tool support for these constraints in an extension of the modeling platform Oryx.

In "Modular protections against non-control data attacks", Cole Schlesinger, Karthik Pattabiraman, Nikhil Swamy, David Walker and Benjamin Zorn introduced YARRA, a conservative extension to C to protect applications from non-control data attacks. YARRA programmers specify their data integrity requirements by declaring critical data types and ascribing these critical types to important data structures.

In "StatVerif: Verification of stateful processes", Myrto Arapinis, Joshua Phillips, Eike Ritter and Mark D. Ryan presented an extension of the ProVerif process calculus with constructs for explicit state, in order to be able to reason about protocols that manipulate global state. They extended the ProVerif compiler to a compiler for StatVerif, and showed the correctness of this compilation.

In "Guiding a general-purpose C verifier to prove cryptographic protocols", François Dupressoir, Andrew D. Gordon, Jan Jürjens and David A. Naumann investigated how to verify security properties of C code for cryptographic protocols by using the general-purpose verifier VCC, thereby proving security theorems in the symbolic model of cryptography. Their techniques include the use of ghost state to attach formal algebraic terms to concrete byte arrays and to detect collisions when two distinct terms map to the same byte array; decoration of a crypto API with contracts based on symbolic terms; and expression of the attacker model in terms of C programs.

We would like to thank the many people who helped make this special issue possible. These include the external reviewers, who put many hours into reviewing submitted papers, providing thoughtful suggestions for improving them. The *JCS* Editors-in-Chief, Pierangela Samarati and Andrew Myers provided guidance through the process.

We hope that you will enjoy this collection of excellent papers!

Michael Backes
Steve Zdancewic
Guest Editors