Guest Editors' Introduction: Special Section on Software Engineering for Secure Systems

Patrick McDaniel and Bashar Nuseibeh, *Member*, *IEEE Computer Society*

THE proliferation of computers in society has meant that organizational and personal assets are increasingly stored and manipulated by software systems. The scale of misuse of these assets has also increased because of their worldwide accessibility through the Internet and the automation of systems. Security is concerned with the prevention of such misuse. While no system can be made completely secure, understanding the context in which a system will be deployed and used, the risks and threats of its misuse, and the systematic development of its software are increasingly recognized as critical to its success.

The cross-fertilization of systems development techniques from software engineering and security engineering offers opportunities to minimize duplication of research efforts in both areas and, more importantly, to bridge gaps in our knowledge of how to develop secure software-intensive systems. The aim of this special issue is to publish novel research work that draws upon software engineering to develop secure systems more effectively. Its scope covers the processes, techniques, technology, people, and knowledge bases that have, or need, the capability to contribute to producing more secure software-intensive systems. In response to the call for papers for this special section, we received 41 submissions, regarding software engineering issues addressing the requirements, design, coding, testing, and maintenance of secure software systems. Each paper was reviewed by at least three expert referees. After two rounds of reviewing, we selected six papers which focus on requirements and design of secure software.

The first two papers address both security and privacy requirements, making use of varying degrees of formalism to represent and analyze those requirements.

"Analyzing Regulatory Rules for Privacy and Security Requirements" by Travis Breaux and Annie Antón addresses the often overwhelming complexity of regulatory requirements of financial, healthcare, and other software. The formalism and process presented glean enforceable security policy directly from the regulatory statutes.

"Privately Finding Specifications" by Westley Weimer and Nina Mishra deals with one of the daunting realities of data sharing, the fact that it is often an all or nothing proposition. This paper describes an attempt to mitigate oversharing in the discovery of software specifications by perturbing program traces. The careful addition of noise into the traces allows specification discovery while preventing the exposure of other sensitive aspects of the program.

The next three papers consider how the design of software systems can be realized through security infrastructure.

"Semantics-Based Design for Secure Web Services" by Massimo Bartoletti, Pierpaolo Degano, Gian Luigi Ferrari, and Roberto Zunino considers how to develop secure Web services by formally reasoning about policy compliance over historical behaviors. In essence, Web services "contract" (compose) with those systems that respect policies of interest, thereby ensuring globally secure behavior.

"Provable Protection against Web Application Vulnerabilities Related to Session Data Dependencies" by Lieven Desmet, Pierre Verbaeten, Wouter Joosen, and Frank Piessens acknowledge recent advances in secure Web application design and development that have made online systems safer. The techniques detailed in this paper prevent misuse of often loosely coupled session dependencies in and among Web applications.

"WASP: Protecting Web Applications Using Positive Tainting and Syntax-Aware Evaluation" by William Halfond, Alessandro Orso, and Panagiotis Manolios presents a novel method for preventing SQL injection attacks—attacks in which the adversary inserts arbitrary database query code into an application by manipulating input strings. The paper uses language techniques to dynamically annotate "trusted" strings, thereby avoiding any use of potentially unsafe strings.

The final paper presents a method of certifying that a software system meets its security requirements.

"Applying Formal Methods to a Certifiably Secure Software System" by Connie Heitmeyer, Myla Archer, Elizabeth Leonard, and John McLean adds to recent advances that are beginning to make this costly and complex process of formal verification tractable. This paper presents a novel certification method that uses formalized security models to construct a mechanized proof of security over a real-world target system.

The papers in this special section demonstrate the strength of research in the area of engineering secure software. If the range and strength of the submissions to the special section are anything to go by, the area is healthy and vibrant and we fully expect many of the submissions that

- *P. McDaniel is with the Department of Computer Science and Engineering, The Pennsylvania State University, 360A Information Sciences and Technology Building, University Park, PA 16802-6823. E-mail: mcdaniel@cse.psu.edu.*
- *B. Nuseibeh is with the Department of Computing, Centre for Research in Computing, The Open University, Walton Hall, Milton Keynes, MK7 6AA, UK. E-mail: b.nuseibeh@open.ac.uk*

For information on obtaining reprints of this article, please send e-mail to: tse@computer.org.

0098-5589/08/$25.00 © 2008 IEEE Published by the IEEE Computer Society

did not make it into this section to appear in regular issues of the journal in the future.

At the International Conference on Software Engineering in 2000, Devanbu and Stubblebine presented a roadmap for research in Software Engineering for Security [2]. In that roadmap, the authors identified the need for research in activities at the early end of the software development life cycle. Seven years on, it is heartening to see the diversity and strength of work in this area, as demonstrated by the papers in this special section. Moreover, the research community continues to make advances in areas only partly covered in this section, including work addressing conceptual [3], social [5], and analytical [4] issues of software engineering of secure systems. Indeed, some of this work has also started to make its way to practitioner communities [1], who increasingly recognize that understanding what security means in the context of the software creation process, and how to achieve it, are central to their goals as practicing software engineers. We invite you, the readers, to participate in the continued development of this research area.

We would like to conclude by thanking the authors of all of the papers who submitted their work to the special section and the 120+ referees who reviewed those papers. Thanks also to former *TSE* Editor-in-Chief Professor John Knight for initiating the special section and setting us on course, current Editor-in-Chief Professor Jeff Kramer for guiding us throughout the submission and review process, and the IEEE Computer Society Editorial Office for their support from start to end.

We hope that you enjoy reading this compilation as much as we did assembling it.

Patrick McDaniel
Bashar Nuseibeh
Guest Editors

REFERENCES

[1] *IEEE Software,* special issue on Security for the Rest of Us, K. Beznosov and B. Chess, eds., Jan./Feb. 2008.

[2] P.T. Devanbu and S. Stubblebine, "Software Engineering for Security: A Roadmap," *Proc. Conf. Future of Software Eng.,* pp. 227-239, 2000.

[3] C.B. Haley, R. Laney, J.D. Moffett, and B. Nuseibeh, "Security Requirements Engineering: A Framework for Representation and Analysis," *IEEE Trans. Software Eng.,* vol. 34, no. 1, pp. 133-153, Jan./Feb. 2008, doi: 10.1109/TSE.2007.70754.

[4] R. De Landtsheer and A. van Lamsweerde, "Reasoning about Confidentiality at Requirements Engineering Time," *Proc. 10th European Software Eng. Conf. and 13th ACM SIGSOFT Int'l Symp. Foundations of Software Eng. (ESEC-FSE '05),* pp. 41-49, 2005.

[5] L. Liu, E. Yu, and J. Mylopoulos, "Security and Privacy Requirements Analysis within a Social Setting," *Proc. 11th IEEE Int'l Requirements Eng. Conf. (RE '03),* pp. 151-161, 2003.

Patrick McDaniel is an associate professor in the Computer Science and Engineering Department at the Pennsylvania State University and co-director of the Systems and Internet Infrastructure Security Laboratory. His research efforts focus primarily on network, telecommunications, and systems security, language-based security, and technical and public policy issues in digital media. He was awarded the US National Science Foundation CAREER Award and has chaired several top conferences in security including, among others, the 2007 and 2008 IEEE Symposium on Security and Privacy and the 2005 USENIX Security Symposium. He is the editor-in-chief of the *ACM Journal Transactions on Internet Technology* (*TOIT*) and serves as associate editor of the journals the *ACM Transactions on Information and System Security* and *IEEE Transactions on Software Engineering*. Prior to pursuing the PhD degree in 1996 at the University of Michigan, he was a software architect and program manager in the telecommunications industry.

Bashar Nuseibeh received the MSc and PhD degrees in software engineering from Imperial College, London, and the First Class Honours BSc degree in computer systems engineering from the University of Sussex, United Kingdom. He is a professor and director of Research in Computing at the Open University (OU) and a visiting professor at Imperial College, London, and the National Institute of Informatics, Japan. Previously, he was a reader at Imperial College, London, and head of its Software Engineering Laboratory. His research interests are in software requirements engineering and design, software process modeling and technology, and technology transfer. He has published more than 100 refereed papers and consulted widely with industry, working with organizations such as the UK National Air Traffic Services (NATS), Texas Instruments, Praxis Critical Systems, Philips Research Labs, and NASA. He has also served as principal investigator on a number of research projects on software engineering, security engineering, and learning technologies. He is editor-in-chief of the *Automated Software Engineering Journal* and an associate editor of the *IEEE Transactions on Software Engineering*, the *Requirements Engineering Journal*, and a number of other international journals. He was a founder and first chairman of the BCS Requirements Engineering Specialist Group (1994-2004) and is currently chair of IFIP Working Group 2.9 (Software Requirements Engineering) and chair of the Steering Committee of the International Conference on Software Engineering (ICSE). He has served as program chair of major conferences in his field, including ASE '98, RE '01, and ICSE '05. He received a Philip Leverhulme Prize (2002), an ICSE "Most Influential Paper" award (2003), a "Best Application Paper" award from the 18th International Conference on Logic Programming (ICLP '02), and a number of other best paper and service awards. He held a Senior Research Fellowship of the Royal Academy of Engineering and The Leverhulme Trust between 2005-2007. He is a fellow of the British Computer Society (FBCS) and the Institution of Engineering and Technology (FIET), a Chartered Engineer (C.Eng.), and a member of the IEEE Computer Society.