Guest Editorial: Security and Dependability in SOA and Business Processes

Ernesto Damiani, *Senior Member*, *IEEE*, Seth Proctor, and Anoop Singhal

THIS special issue presents recent research results in a field of research that is itself rather new. When Service Oriented Architectures (SOA) came of age, no specific security technology for web services was available and transport protocols security mechanisms were used instead. For instance, web services message confidentiality was achieved using transport security protocols like SSL and HTTPS. Web services that needed authentication used transport authentication (i.e., the Basic or Digest HTTP authentication mechanisms) or certificate-based schemes. When the research community started to address the problem of web service security, we had to recognize that many of the features that make web services attractive (above all, composition and open service-to-service invocation) conflicted with traditional security models and solutions. So, it was back to the drawing board for many of us. Meanwhile, securing web services looked to many practitioners more like an art than a science.

A major problem that surfaced early was supporting authenticity of service invocations across compositions. Indeed, weak authentication chains were at the basis of many early attacks to services. Today, SOAP headers support SOAP-specific security mechanisms that aim to achieve a) end-to-end security along the chain of intermediaries leading to a SOAP web service and b) full independence from the security mechanisms of transport protocols. In this special issue, the paper "Two-Dimensional Trust Rating Aggregations in Service-Oriented Applications" by Yan Wang and Lei Li provides an up-to-date view of the crucial problem of aggregating trust levels within composite services and business processes.

Early debate on web service security also brought forward the idea of supporting message-level security using SOAP headers.

Besides authentication, SOAP headers have been used since then to support a number of security mechanisms. Headers can carry encryption metadata, ensuring confidentiality of a SOAP message, or information on a digital signature scheme according to the XML Signature standard, ensuring that SOAP messages have originated from the appropriate client and were not modified in transit. Also, SOAP headers can be used to return to clients a security token to be used in future calls to the service.

These security mechanisms are now well understood, and research is focusing on the performance problems posed by processing SOAP security headers. In this special issue, the paper "Server-Side Streaming Processing of WS-Security," by Nils Gruschka et al., paves the way to efficiently enforcing SOAP security.

A distinct though closely related issue is using XML-based languages to express access permissions to web services. More than 11 years ago, one of us (E. Damiani) wrote and sent to the W3C mailing list an "XML access control manifesto" stating that "Using XML to express access and usage policies will allow for naturally expressing such policies organization-wide (associating a policy to an XML schema) and site-wide (associating a policy to a single XML document). Like usual metadata, access and usage policies expressed this way are both machine and human-readable; moreover, they can be transferred together with data, and processed via standard enforcement engines."

In the following years, much work research was devoted to developing XML-based policy languages and models. The XACML (eXtensible Access Control Markup Language) specification emerged, defining a declarative access control policy language implemented in XML and a processing model describing how to interpret the policies.

While SOAP web services were considered a natural target for XACML policies from the very beginning of XACML standardization, a major problem when using XACML to state access control policies for SOAP web services is the naming of resources, as SOAP data objects are typically not made available through a URI.

The "Web Services Profile" of XACML (WS-XACML), written by Anne Anderson, bridged this gap by proposing XACML-based formats for authorization and privacy policies for web services. Today, XACML is still an important reference for research. In this special issue, the paper "Runtime Administration of an RBAC Profile for XACML," by Xu Min et al., describes a solution for efficient administration of role-based access control policies using XACML, while the paper "Adaptive Reordering and Clustering Based Framework for Efficient XACML Policy Evaluation," by Mohamed Shehab et al., describes an innovative framework for efficient evaluation of XACML policies. The Web Services Security specification (WS-Security) can also be regarded as a development of the idea of using SOAP headers to carry security-related information. It is closely related to the WS-Policy specification, that, in turn, develops the idea of a machine-readable format for access control policies.

- *E. Damiani is with the Department of Information Technology, Università degli Studi di Milano, Italy. E-mail: ernesto.damiani@unimi.it.*
- *S. Proctor is with NimbusDB. E-mail: stp@alumni.brown.edu.*
- *A. Singhal is with the Computer Security Division, National Institute of Standards and Technology. E-mail: anoop.singhal@nist.gov.*

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

1939-1374/11/$26.00 © 2011 IEEE Published by the IEEE Computer Society

Regardless of the access control language used, research on controlling access to web services needs to address a number of hard problems concerning policy representation and enforcement. In the paper "Security Policy Composition for Composite Web Services," Fumiko Satoh and Takehiro Tokuda provide some interesting results on efficiently computing the composition of policies regulating access to services.

Dependability is another crucial property of service-oriented applications. While design patterns for dependable atomic services have long been proposed, ensuring the dependability of a service composition is a much harder problem. The paper "Dependability and Rollback Recovery for Composite Web Services," by Houwayda Elfawal Mansour and Tharam Dillon, provides a new, promising solution to this difficult problem.

Obstacles on the way toward achieving certifiably high levels of assurance for service-based applications are still formidable, especially when some services composing a business process are outsourced and, therefore, not under the control of the business process owner. The paper "A Data Assurance Policy Specification and Enforcement Framework for Outsourced Services," by Jun Li et al., deals with the assurance problem, introducing the notion of a data assurance policy for outsourced services and processes.

The security and dependability challenges presented by web services approaches are still formidable. However, the papers collected in this special issue show that some building blocks are now firmly in place. Thanks to the reseach community, securing service-based applications has become closer to an engineering discipline than to an art.

ACKNOWLEDGMENTS

The guest editors wish to thank the anonymous referees for their great work in putting together this special issue.

Ernesto Damiani received the MS degree in computer engineering from Università di Pavia, Italy, and the PhD degree in computer science from the Università degli Studi di Milano, Italy. He is currently a full professor in the Department of Information Technology at the Università degli Studi di Milano, and the head of the university's PhD program in computer science. Dr. Damiani has held visiting positions at many institutions worldwide, including George Mason University, Virginia; La Trobe University, Melbourne, Australia; and INSA-Lyon, France. His research interests include business process representation and metrics, secure service-oriented architectures, and software process engineering. He has published more than 200 scientific papers and several books. Dr. Damiani was the coproposer of the XACML standardization group. He is the vice-chair of the IEEE technical committee on Industrial Informatics, the chair of the IFIP WG on Data Semantics (WG 2.6), and the vice-chair of the IFIP WG 2.13 on Open Source Software. He is a senior member of the IEEE.

Seth Proctor received the PhD degree in computer science from Brown University. He was a senior software engineer at Nokia and he later worked as a researcher at Sun Microsystems Laboratories, where he took part in the development of the XACML specification. He is currently a developer at NimbusDB.

Anoop Singhal received the PhD degree in computer science from Ohio State University, Columbus, the MS degree in computer science from Ohio State University, Columbus, and the BTech degree in electrical engineering from the Indian Institute of Technology, Delhii. He is currently a senior scientist at the National Institute of Standards and Technology, Computer Security Division. His research interests include network security, intrusion detection, data mining, and web services security.