Web and Information Security

Elena Ferrari
University of Insubria at Como, Italy

Bhavani Thuraisingham
University of Texas at Dallas, USA

IRM Press
Publisher of innovative scholarly and professional information technology titles in the cyberage
Hershey • London • Melbourne • Singapore

Acquisitions Editor:	Michelle Potter
Development Editor:	Kristin Roth
Senior Managing Editor:	Amanda Appicello
Managing Editor:	Jennifer Neidig
Copy Editor:	April Schmidt
Typesetter:	Jennifer Neidig
Cover Design:	Lisa Tosheff
Printed at:	Yurchak Printing Inc.

Published in the United States of America by
IRM Press (an imprint of Idea Group Inc.)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033-1240
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@idea-group.com
Web site: http://www.irm-press.com

and in the United Kingdom by
IRM Press (an imprint of Idea Group Inc.)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 3313
Web site: http://www.eurospan.co.uk

Library of Congress Cataloging-in-Publication Data

Web and information security / Elena Ferrari and Bhavani Thuraisingham, editors.
p. cm.
Summary: "This book covers basic concepts of web and information system security and provides new insights into the semantic web field and its related security challenges"--Provided by publisher.
Includes bibliographical references and index.
ISBN 1-59140-588-2 (hardcover) -- ISBN 1-59140-589-0 (softcover) -- ISBN 1-59140-590-4 (ebook)
1. Computer networks--Security measures. 2. Web sites--Security measures. 3. Computer security. 4. Semantic Web. I. Ferrari, Elena, 1968- II. Thuraisingham, Bhavani M.
TK5105.59.W42 2006
005.8--dc22
2005020191

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Web and Information Security

Table of Contents

Section II: Policy Management and Web Security

Section III: Security for Emerging Applications

Preface

Recent developments in information systems technologies have resulted in computerizing many applications in various business areas. Data have become a critical resource in many organizations; therefore, efficient access to data, sharing data, extracting information from data, and making use of information has become an urgent need. As a result, there have been many efforts not only on integrating the various data sources scattered across several sites but also on extracting information from these databases in the form of patterns and trends. These data sources may be databases managed by database management systems, or they could be data warehoused in a repository from multiple data sources.

The advent of the World Wide Web (WWW) in the mid-1990s has resulted in an even greater demand for managing data, information, and knowledge effectively. There is now so much data on the Web that managing it with conventional tools is becoming almost impossible. New tools and techniques are needed to effectively manage these data. Therefore, to provide interoperability as well as warehousing between the multiple data sources and systems, and to extract information from the databases and warehouses on the Web, various tools are being developed.

As the demand for data and information management increases, there is also a critical need for maintaining the security of the data sources, applications, and information systems. Data and information have to be protected from unauthorized access as well as from malicious corruption. With the advent of the Web, it is even more important to protect the data and information as numerous individuals now have access to these data and information. Therefore, we need effective mechanisms for securing access to data and applications.

Due to the numerous developments in Web and information systems security and the great demand for security in emerging systems and applications, we held a workshop in this field at the IEEE Institute for Electrical and Electronics Engineers Computer Society's COMPSAC (Computer Systems and Applications) Conference in August 2002 at Oxford, UK. Subsequently, we decided to edit a book in the field due to the numerous requests we received from our colleagues. This edited collection of papers consists of vastly enhanced versions of some of the papers that were presented at the workshop, together with several additional papers on state-of-the-art topics such as Semantic Web security and sensor information security. We will first review the developments in *Web and Information Systems Security* and then discus the contents of the book.

Developments in Web and Information Systems Security

Web and Information Systems security have roots in database and applications security. Initial developments in database security began in the 1970s. For example, as part of the research on System R at IBM Almaden Research Center, there was a lot of work on access control for relational database systems. About the same time, some early work on multi-level secure database management systems (MLS/DBMSs) was reported.

However, it was only after the Air Force Summer Study in 1982 that much of the developments on secure database systems began. There were the early prototypes based on the integrity lock mechanisms developed at the MITRE Corporation. Later in the mid-1980s, pioneering research was carried out at SRI International and Honeywell, Inc. on systems such as SeaView and LOCK Data Views. Some of the technologies developed by these research efforts were transferred to commercial products by corporations such as Oracle, Sybase, and Informix.

The research in the mid-1980s also resulted in exploring some new areas such as the inference problem, secure object database systems, and secure distributed database systems. In fact, Dr. John Campbell of the National Security Agency stated in 1990 that one of the important developments in database security was the work by Thuraisingham on the unsolvability of the inference problem. This research then led the way to examine various classes of the inference problem. Throughout the early 1990s, there were many efforts reported on these new types of secure database systems by researchers at organizations such as the MITRE Corporation, Naval Research Laboratory, the University of Milano, and George Mason University. In addition, much work was also carried out on secure transactions processing.

In the mid-1990s with the advent of the Web, there were many new directions for secure data management and applications research. These included secure workflow systems, secure digital libraries, Web security, and secure data warehouses. New technologies, such as data mining, exacerbate the inference problem as even naive users could use data mining tools and infer sensitive information. Closely related to the inference problem is the privacy problem where users associate pieces of public data together and deduce private information. Data mining also exacerbates the privacy problem. However, data mining is also a very important technique for solving many security problems such as intrusion detection and auditing. Therefore, the challenge is to carry out data mining but, at the same time, ensure the inference problem is limited. Developments in distributed object systems and e-commerce applications resulted in developments in secure distributed object systems and secure e-commerce applications. In addition, access control has received a lot of attention especially in the area of role-based access control (RBAC).

Recently, there have been numerous developments in data and applications security. Everyday, we are seeing developments in Web data management. For example, standards such as XML (eXtensible Markup Language) and RDF (Resource Description Framework) are emerging. Security for these Web standards has to be examined. Also, Web services and the Semantic Web are becoming extremely popular; therefore, we need to examine the related security issues. Security is being examined for new application areas such as knowledge management, peer-to-peer computing, and sensor data management. For example, in the case of knowledge management applications, it is important to protect the intellectual property of an organization. Privacy should be an important consideration when managing surveillance data emanating from sensors. Peer-to-peer computing has received a lot of attention recently. There are numerous security issues for such systems, including secure information sharing and collaboration. Furthermore, data are no longer in structured databases only. Data could be streams emanating from sensors and other sources as well as text, images, and video. Security for such data has not yet received much attention. Finally, one has to make tradeoffs between security, data quality, and real-time processing. In summary, as new technologies emerge, there are many security issues that need to be examined. We have made much progress in data and applications security in the last three decades, and the chapters in this book discuss some of the state-of-the-art developments.

Aims of This Book

This book provides some of the key developments, directions, and challenges for securing the Semantic Web, enforcing security policies, as well as securing some of the emerging systems such as multimedia and collaborative systems. It could be used as a reference book for senior undergraduate or graduate courses in information security which have a special focus on Web security. It is also useful for technologists, managers, and developers who want to know more about emerging security technologies. It is written by experts in the field of information security, Semantic Web, multimedia systems, group collaboration systems, and data mining systems.

Organization of This Book

This book is divided into three sections, each addressing a state-of-the-art topic in Web and information systems security. They are as follows: *Securing the Semantic Web*, *Policy Management and Web Security*, and *Security for Emerging Applications*. We discuss the trends in each topic and summarize the chapters.

Section I: Securing the Semantic Web

Semantic Web is essentially about machine-understandable Web pages and was conceived by Tim Berners-Lee. The World Wide Consortium has made major developments on the Semantic Web. Current challenges include securing the Semantic Web as well as making the Semantic Web more intelligent.

Section I consists of five chapters addressing various aspects of securing the Semantic Web. The first chapter, "Creating a Policy-Aware Web: Discretionary, Rule-Based Access for the World Wide Web", by Weitzner, Hendler, Berners-Lee, and Connolly, discusses how to define and enforce security policies for the Semantic Web. It focuses on rule-based policies for the Semantic Web. The second chapter, "Web Services Security", by Garciá, Patón, and Velthius, describes issues on securing Web services. In particular, it focuses on areas that need to be standardized. The third chapter, "Policies for Web Security Services", by Stoupa and Vakali, focuses on defining and enforcing security policies for Web services. In particular, it analyzes the various policies implemented by Web services in the areas of confidentiality, authentication, non-repudiation, and integrity and access control. The fourth chapter, "Data Confidentiality on the Semantic Web: Is There an Inference Problem?," by Farkas, shows how the inference problem can be handled in the Semantic Web. It focuses on the inference problem resulting from RDF specifications as well as Ontology specifications. The fifth and final chapter in this section, titled "Secure Semantic Grids", by Thuraisingham and Khan, shows how the concepts from secure Semantic Web and secure grid can be integrated to secure the semantic grid.

Section II: Policy Management and Web Security

Since the development of access control policies in the 1970s, numerous developments have been made on policy specification and management for secure systems. Section II consists of five chapters focusing on various policy issues for Web-based information systems. The first chapter, "Web Content Filtering", by Bertino, Ferrari, and Perego, discusses issues on developing filters that remove unwanted information on the Web. In particular, it describes how users can be prevented from accessing harmful content. The second chapter, "Sanitization and Anonymization of Document Repositories", by Saygin, Hakkani-Tür, and Tür, describes techniques for sanitizing document repositories. Its main focus is on privacy for text documents. The third chapter, "Integrating Access Policies into the Development Process of Hypermedia Web Systems", by Díaz, Sanz, Montero, and Aedo, describes how access control policies may be specified and enforced in a hypermedia-based system. The two ideas proposed are the use of high-level access control models and the inclusion of access control in the life cycle of hypermedia applications. The fourth chapter, "Policy-Based Management of Web and Information Systems Security: An Emerging Technology", by Pérez, Clemente, and Skarmeta, describes how various policies may be used to manage and administer Web-based systems. In particular, they provide a system view of the network and its services and discuss policy management in such an environment. Finally, the fifth and last chapter of this section, titled "Chinese Wall Security Policy Model: Granular Computing on DAC Model", by Lin, argues that the Chinese Wall model cannot only be used for mandatory access control but also for discretionary access control. It goes on to give mathematical arguments to support the thesis.

Section III: Security for Emerging Applications

Recently, there have been numerous developments on incorporating security into emerging systems and applications, including data warehouses, data mining systems, multimedia systems, sensor systems, and collaborative systems. Part III of this book, consisting of four chapters, focuses on incorporating security into some of these emerging systems. The first chapter, "A Multimedia-Based Threat Management and Information Security Framework", by Joshi, Shyu, Chen, Aref, and Ghafoor, describes security for multimedia systems. It focuses on integrating disparate components to support large-scale multimedia applications and discusses threat management in such an environment. The second chapter, "Framework for Secure Information Management in Critical Systems", by Kannan, Iyengar, and Durresi, discusses security for sensor information systems. It focuses on confidentiality, anonymity, and integrity and discusses the tradeoffs between these features. The third chapter, “Trustworthy Data Sharing in Collaborative Pervasive Computing Environments”, by Yau, describes security for group communication and collaboration. It focuses on flexible data sharing as well as on effective data replication mechanisms. The fourth and final chapter, “Privacy-Preserving Data Mining on the Web: Foundations and Techniques”, by Oliveira and Zaïane, describes how one can carry out data mining and, at the same time, maintain privacy. It stresses that understanding privacy is important in order to develop effective solutions for privacy preserving data mining.

Elena Ferrari, University of Insubria at Como, Italy
Bhavani Thuraisingham, University of Texas at Dallas, USA

May 2005

Acknowledgments

The editors would like to thank all the people that made the successful completion of this project possible. First, we would like to thank the publishing team at Idea Group Publishing. In particular, we would like to thank Mehdi Khosrow-Pour who gave us the opportunity to edit this book, and Jan Travers, Michele Rossi, Kristin Roth, Renée Davies, Amanda Appicello, Jennifer Neidig, April Schmidt, and Lisa Tosheff for their constant support throughout the whole process.

We also want to express our gratitude to the authors of the chapters for their insights and excellent contribution to this book. Most of them also served as referees for chapters written by other authors. We wish to thank all of them for their constructive and comprehensive reviews.

Section I

Securing the Semantic Web

Chapter I

Creating a Policy-Aware Web: Discretionary, Rule-Based Access for the World Wide Web

Daniel J. Weitzner, Massachusetts Institute of Technology, USA

Jim Hendler, University of Maryland, USA

Tim Berners-Lee, Massachusetts Institute of Technology, USA

Dan Connolly, Massachusetts Institute of Technology, USA

Abstract

In this chapter, we describe the motivations for, and development of, a rule-based policy management system that can be deployed in the open and distributed milieu of the World Wide Web. We discuss the necessary features of such a system in creating a "Policy Aware" infrastructure for the Web and argue for the necessity of such infrastructure. We then show how the integration of a Semantic Web rules language (N3) with a theorem prover designed for the Web (Cwm) makes it possible to use the Hypertext Transport Protocol (HTTP) to provide a scalable mechanism

for the exchange of rules and, eventually, proofs for access control on the Web. We also discuss which aspects of the Policy Aware Web *are enabled by the current mechanism and describe future research needed to make the widespread deployment of rules and proofs on the Web a reality.*

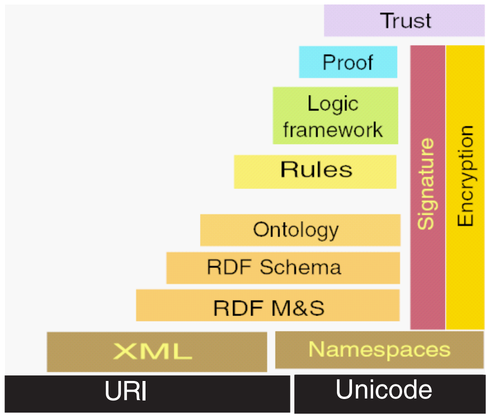
Introduction

Inflexible and simplistic security and access control for the decentralized environment of the World Wide Web have hampered the full development of the Web as a social information space because, in general, the lack of sufficiently sophisticated information controls leads to unwillingness to share information. This problem is greatly exacerbated when information must be shared between parties that do not have pre-existing information-sharing policies and where the "granularity" of the information to be shared is coarse—that is, where access is granted to an entire Web site or data resource because policy control mechanisms for access at a finer-grained level are not available. Even large intranets and controlled-access Webs face these problems as the amount of information and the number of information seekers grow. Thus, despite ever-greater amounts of useful information residing on the Web in a machine-retrieval form, reluctance to share that information remains and is likely to increase.

In this chapter, we will argue that a new generation of *Policy-Aware* Web technology can hold the key for providing open, distributed, and scalable information access on the World Wide Web. Our approach provides for the publication of declarative access policies in a way that allows significant transparency for sharing among partners without requiring pre-agreement. In addition, greater control over information release can be placed in the hands of the information owner, allowing discretionary (rather than mandatory) access control to flourish.

The technical foundation of our work focuses on developing and deploying the upper layers of the "Semantic Web layer-cake" (Figure 1, based on Berners-Lee, 2000; Swartz & Hendler, 2001) in order to enable Policy-Aware infrastructure. The ambition of the Semantic Web is to enable people to have richer interactions with information online through structured, machine-assisted integration of data from all around the Web (Berners-Lee, Hendler, & Lassila, 2001). We will show that it is possible to deploy rules in a distributed and open

Figure 1. Semantic Web Layer Cake ca. 2002



system, and to produce and exchange proofs based on these rules in a scalable way. These techniques, properly applied by taking crucial Web architecture issues into account, will extend Semantic Web technology to allow information resources on the World Wide Web to carry access policies that allow a wide dissemination of information without sacrificing individual privacy concerns.

The ultimate success of the Semantic Web, however, will depend as much on the *social* conditions of its use as on the underlying technology itself. Much of the power of the Semantic Web lies in its ability to help people share information more richly and to discover subtle information linkages across the Web that are not visible in today's relatively flat online information environment. However, people will not share information freely in an environment that is threatening or antithetical to basic social needs such as privacy, security, the free flow of information, and ability to exercise their intellectual property rights as they chose. Though today's Web falls short in many of these areas, the descriptive and logical functions of the Semantic Web can offer the ability to help people manage their social relationship online, in addition to just managing the traditional information content found on the Web today. We describe here the framework for, and first steps toward, a *policy aware* Web.

As an integral part of the Semantic Web, policy-aware infrastructure can give users greater transparency in their online interactions, help both people and machines to play by the rules relevant to social interactions in which they participate, and provide some accountability where rules are broken. The Policy-Aware Web is the logical continuation of the "user empowering" features of the Web that have, in the Web's first decade, been critical in shaping the delicate relationship between Web technology and the surrounding legal environment (Berman & Weitzner, 1995).

In this chapter, our primary focus will be on the use of Semantic Web technologies to provide a rule-based access mechanism in a style that is consistent with current and expected future Web Architecture. First, however, we describe what we mean by policy awareness and the needs of bringing it to the online world.

Being Policy Aware

By any measure, today's World Wide Web has been extraordinarily successful at meeting certain social goals and rather disappointing at others. The Web has enhanced dissemination of, and access to, information in both commercial and non-commercial contexts. We have seen great ease of publishing relative to mass media and constantly improving search and discovery. The Web has even provided relatively robust responses to the great diversity of opinion about what constitutes good, bad, moral, immoral, legal, and illegal content (cf. Reno vs. ACLU, 1977). Yet for all of the Web's success at meeting communication and information exchange goals, it has failed in equal measure at satisfying other critical policy requirements such as privacy protection, a balanced approach to intellectual property rights, and basic security and access control needs. We worry about these problems not only because they implicate fundamental human rights, but also because the failure to solve them renders this medium that we all care about that much poorer and causes people to feel alienated in their online interactions, even as they appreciate the unprecedented benefits of the Web.

As these problems fall into the category of law and public policy, the general impulse is to look to the law to solve them. Law is certainly a necessary part of making the Web a humane environment, but it is not sufficient alone. For as much as there are real deficiencies in the laws that govern online interactions, the absence of technical capacity to share basic context information between users and services providers, and among users, is a fundamental impediment to the Web being an environment in which people will feel comfortable and confident to conduct a full range of human activities. Indeed, the focus on law as a solution to the policy-related problems on the Web risks obscuring the deep technical and functional gaps that prevent us from having normal social interactions online.

To illustrate these gaps, consider the differences in policy awareness regarding the flow of sensitive personal information between browsing in your local library and browsing an online digital library repository. In either case, your browsing habits may be tracked, perhaps even in a way that associates your name with the information collected. The similarity ends there because off-line, if an overeager librarian follows you from aisle to aisle looking at which books you pick up and whether you open the pages or not, you would both know that this was happening and have a variety of understated but clear techniques for stopping the behavior or at least making your displeasure known. Our sense of vision (to notice the snooping) and mastery of simple gestures (the quizzical or displeasing look over the shoulder) help us to be aware of and resolve this awkward situation. Only in the oddest of circumstances would recourse to law be required or even useful. A simple exchange of social clues would more than likely solve the problem.

When this scenario is replayed in an online library, however, the user doing the browsing is at a distinct disadvantage. First, it is quite unlikely that the online browser will even be aware of the tracking behavior (or lack of it) unless she has found a privacy policy associated with the site and managed to read and understand it. Even with that, the policy is likely to describe what the site might do, not what actually happens in the case of a given browser on a given visit. Second, even if the online library browser ascertained that unwanted tracking was occurring, what could she do? We have no online equivalent of shooting the snooper a dirty look or sneaking down another aisle.

This gap between what is possible in the online and off-line environment has a critical impact on the degree which people feel comfortable interacting online. As the library example illustrates, in most human interaction, we rely on various feedback loops to establish what is acceptable versus unacceptable behavior. Online environments that lack the channels for such feedback thus need to replace these mechanisms with other, more Web appropriate ways of maintaining our mastery over our personal information space. In order to make the Web a more socially-rich environment, we can take advantage of the rich representational framework offered by the Semantic Web to help people manage not just the traditional Web content but also the social context and cues around any information-related activity.

Consider the simple desire to share photographs among friends. Off-line, if you want to share a picture with a friend or colleague, you have an easy way to give them the picture, and it is very likely that the context of that interaction and your relationship will give the recipient of the photo a pretty good clue about the social rules to be associated with the use and sharing of that picture. Of course, today we can e-mail pictures around, and many of the same social conventions are likely to apply. But try to use the Web to share pictures with the informally-defined communities in which we all participate, and problems soon emerge. While the Web allows us to access and transport pictures all around the world to hundreds of millions of potential recipients, the inability to specify even very simple rules for sharing information forces us into an uncomfortably inflexible set of choices: share with everyone, share with no one, or engage in the arduous task of managing access via IP addresses or assigning names and passwords.

The lack of *policy awareness* in today's Web infrastructure makes it difficult for people to function as they normally would in informal or ad hoc communities. Thus, policy awareness is a property of the Semantic Web that will provide users with readily accessible and understandable views of the policies associated with resources, make compliance with stated rules easy, or at least generally easier than not complying, and provide accountability when rules are intentionally or accidentally broken. So, in building Policy-Aware services, we seek to meet the following requirements:

- **Transparency:** Both people and machines need to be able to discover, interpret, and form common understandings of the social rules under which any given resource seeks to operate. Can it be shared, copied, commented upon, made public, sold, and so forth? Encoding social rules in the formal mechanisms described below will provide a level of transparency currently unavailable on today's Web (Weitzner, 2004). What remains is to develop the social practice of using these mechanisms in consistent ways to communicate about social context and expectations. Related work has been done in the context of existing Web standards such as the Platform for Privacy Preferences (P3P) and XML markup languages such as SAML, EPAL, and XACML. However, research is still required to enable the development of local community-specific policy description frameworks and tools to help users evaluate policy rules, especially when various rule sets interact.

- **Compliance mechanisms:** We would like it to be just as easy to comply with rules expressed in a policy-aware environment as it is to use the Web today. Thus, most users must be largely unaware of the underlying formalisms in which the policies are expressed and maintained, and mechanisms built into the structure of the Web (protocols, browsers, etc.) should support the policies thus expressed. The mechanism we describe in this chapter uses rules and transportable proofs as the communications channel through which the user establishes compliance with a given rule set with the discovery and use of the rules built into the Web infrastructure. Expression of social rules in a formal, machine-readable manner will enable end-user software (including browsers and other user agents) to make it easier for users to comply with the rules of the environment in which they participate.
- **Accountability:** Rules, no matter how well described or carefully enforced, may be broken. Whether the breach is inadvertent or intentional, a policy-aware environment will help participants to spot and track infractions. In some cases, there may have been a misunderstanding or inadvertent error. Or, in large user communities such as the Web, it is certainly possible that the breach was malicious. The individuals and communities involved will respond in different ways depending on the social and legal context of the breach. Policy awareness seeks to identify rule violation with adequate accountability and context sensitivity so that those involved can take whatever action is appropriate.

Based on these principles, a key difference between policy-aware access control, of the sort that we describe in this chapter, and traditional access control approaches, developed in the computer security and cryptography community, is that we stress description over enforcement. In current systems, often the description of the policies is intertwined with the enforcement thereof. Cryptographic enforcement mechanisms generally require a high degree of pre-coordination on policy terms and demand that users and system administrators bear the costs of maintaining a local public key management infrastructure. While these costs may be acceptable to certain environments which must protect high value assets (commercial financial transactions or intelligence information, for example), they are entirely beyond the means of small ad hoc communities. In these cases, most users will continue to live with virtually no access control mechanisms at all. Our aim is thus to give people the ability to have highly descriptive security policies with a relatively low enforcement burden placed on the individual Web client. Hence, we concentrate our energies on describing access control policies and providing the tools to enable policy-aware systems to assess compliance with rules based on good faith assertions from all involved. The policy-aware approach can work well with more robust cryptographically-enforced security as well, as we will describe later in this chapter, but our current emphasis is at the high description end of the spectrum, rather than at the high enforcement end.

One notable piece of past work in the area of highly descriptive access on the Web is that of the REI system (Kagal et al., 2004). REI extends a rule-based policy mechanism developed for distributed processing applications. REI is based on an agent-based computing approach, in which agents (realized primarily as Web services) are able to control access and information sharing via policies encoded in OWL ontologies. Our work is closely related to ideas in REI but is focused on going beyond their multi-agent, service-based paradigm and building rule-based access into the Web protocols themselves, with an emphasis on application to the decentralized environment of the Web.

Rule-Based Access and the World Wide Web

Research in the security area has recently been exploring mechanisms that allow the requirements above to be realized by the use of "rule-based" access policies, shifting away from the identity- and role-based mechanisms that are the primary mechanisms used on the Web today (where any access control is used at all). Our work focuses on extending rule-based access to be used in the open and distributed World Wide Web, which is necessary for achieving the policy-awareness goals described above. In this section, we provide some background on past work and define the goal of our research, as well as identify some of the key pieces of work that we build on.

Most Web access today is performed using identity-based approaches (Shamir, 1985) where access to all or some of the data is granted based on pre-existing agreements negotiated between the data owner and those accessing the data resource. A simple example of this is password-based access to a protected Web site—a user who identifies him or herself by providing the correct login/password combination is allowed in, others are not. Identity schemes are also used in many database systems for both online and off-line access, with more recent work focused on using public key certificates, rather than passwords, to add more security (cf. Boneh & Franklin, 2001). Role-based access (cf. Ferraiolo, Kuhn, & Chandramouli, 2003) is similar to identification-based access, except that instead of identifying a particular user, an access policy is created to allow users of a particular class (i.e., those who play some role) to access various parts of the data. Thus, for example, the World Wide Web Consortium (W3C) Web site has an access policy that (simplifying somewhat) allows users to be assigned to three classes by their roles—*team,* which has access to all files; *member,* which has access to all files accept those marked *team*; and *public,* which has access to all files except those marked team or member.

There are several problems with identity- and role-based schemes. First, in most cases the classes must be defined in advance. Creating a temporary class is difficult, if not impossible, in most implementations of these policies. For example, in preparing this chapter one author, Hendler, needed access to a W3C document that was labeled *team*, but he only had *member* access rights. Giving Hendler *team* access would have meant letting him see other documents he did not have the right to view. Moving the document to *member* would have risked letting it be seen by others who served the same role as Hendler but were not entitled to see this particular document. In the end, moving the document to a different site where we could set up a temporary (password-based) scheme was more trouble than it was worth, and instead we had to resort to e-mailing the document to each other (a workaround which bypassed the entire security system).

A second problem with these schemes is that they tend to be difficult to set up in a fine-grained way as Web-based schemes generally work at the file-directory level. It is difficult, for example, to give someone access to a part of your page or to particular data in a specific context.[2] Our goal is to be able to write rules that describe policies at the level of individual URIs, thus grounding the system in the smallest externally nameable Web resources. Our decision to base our approach on RDF, rather than XML, is largely based on the fact that RDF assigns individual URIs to instances and classes, seemingly making it ideal for this purpose. (It is worth noting, however, that current Web protocols still return an entire document, rather than the individual named entity, when URIs containing "fragIDs" are used. It is our hope that RDF query languages currently under development will allow delivery of finer-grained query responses from RDF stores, thus helping to alleviate this problem. For non-text resources such as individual photos within a photo collection, current Web protocols allow appropriate experimentation with finer-grained access.)

A third limitation of these schemes is that it is usually extremely difficult to have precise access change over time. For example, a better solution to the access problem described previously would have been to temporarily create a "*team+hendler*" role and to have the document in question be limited to *team+hendler* until some specific date, at which time the new role could go away, and the document could revert to its previous state. Defining time-sensitive rules is difficult in role-based schemes.

The ability to specify access policies that do not have to be defined in advance, have fine grained access, and allow fairly dynamic change is a current focus of research in the database (Kyte, 2000), Programming Language (Pandey & Hashli, 1999), Operating System (Ott, 2001), Artificial Intelligence (Barbour, 2002), and multiparty security (cf. the PORTIA, SDSI, and SPKI projects) areas. This work largely focuses on a switch from role-based authentication to what is known as *rule-based* access policies (cf. Didriksen, 1997), an approach which has been gaining popularity since the late 1990s. In rule-based access, a declarative set of rules is used to define finer-grained access to resources with requests for data providing a "demonstration" that they satisfy the policy encoded in the rules. The demonstration of meeting these rules can be fairly simple—for example, most commercial implementations of rule-based access have only simple antecedents that can match information in (public key) certificates to features in the data.

To date rule-based access has been primarily associated with Mandatory Access Control (MAC) systems, especially those used to provide multi-level access to documents. MAC systems are those where the owner of the information does not get to control protection decisions, but rather the system is designed to enforce *a priori* protection decisions (i.e., the system enforces the security policy possibly over the wishes or intentions of the object owner). In these systems, now in common use in both industrial and government applications, every "information object" is tagged with a sensitivity level, and every "subject" (generally a process which can cause information to flow – i.e., something which can remove data objects from the system) is also given a tag. A lattice of subject/object pairs is used and a simple set of rules implemented that will only allow a subject access to an object if its tag has a position in the lattice that is equal to or higher than that of the object.

Rule-based systems have been less successful, however, in Discretionary Access Control (DAC) systems, where the information owner can locally determine the access policy. The reason for this is that rule-based access generally requires the subject to "prove" they have access and for the objects in the system to have a finer-grained control of the information than is typical in MAC schemes.

For example, the provider of some data may wish to have an access policy that would match a rule such as:

```
User  can-access  DATA.Associated-record.{decision,signees}
  IF   User.attribute-certificate:originator= W3C
    And DATA.type-designator= Member
      And Employer=User.employer
        And Employer.member:status = current
```

That is, users who are authorized by the W3C (i.e., providing an attribute certificate[3] signed by an authorized W3C entity) can see particular documents (in this case, the decision and who signed it) for those data elements which are allowed to be seen by those in the "member" group and where the employer associated with the user is identified as being a current W3C member organization.

It is this ability to create a discretionary, rule-based access control on the Web that we are trying to achieve—that is, we believe such rule-based mechanisms will be a necessary component of the *policy-aware Web,* as the ability to control access will be an integral part of the privacy and sharing controls described previously. Our goal, therefore, is to show how rule-based access methods can be brought to the Web using the same principles of openness, distribution, and scalability that have allowed the Web to grow into the pervasive application that it is today.

Technical Challenges

There are many challenges inherent in bringing rule-based, discretionary access control to the Web. Contrast the Web access problem to the typical database (or OS) access issues, and it becomes clear why this is so:

1. Current rule-based schemes use specialized access control languages generally designed to work in a specific application. Such proprietary approaches rarely work on the Web, due to the need for openness and

shared use. If my application cannot read your rules, or yours cannot read mine, then we do not get interoperability. Indeed, it is not enough to have a standard for writing the rules; it is critical that the mechanisms by which the rules governing access to an entity can be expressed in a flexible manner, discovered easily, and applied in a reliable manner, preferably within the scope of the Hypertext Transfer Protocol (HTTP) itself.

2. In a closed, controlled system, a pre-defined set of subject tags checked against a predefined set of object tags is sufficient—in fact, this is why rule-based MAC has become viable for many organizations. On the Web, however, there is no simple set of tags that will be sufficient for all applications in all domains. Instead, a mechanism must be provided that can evaluate the rules in the policy against information provided by the subject. This information can be in the form of a signed access certificate or other such identity provider, a Web-based proof, or some combination thereof.
3. When access certificates and other such identifiers are not sufficient, there must be a general mechanism for providing a *proof* that one system is allowed to access the information in some resource using published rules (Bauer, 2003). On the Web, the subject must provide some sort of grounded and authenticatible proof that an object's access policy can be met, and the proof must be exchanged using Web protocols. In addition, there must be a mechanism by which the system receiving the proof can check its correctness with respect to only those rules of logic that it accepts. On the Web, we cannot assume that every user will employ the same piece of proof-checking software, so a set of standards is required be sure that all participants evaluate proofs on the Semantic Web in a consistent manner. Some tools may develop that only use a few simple rules (perhaps limiting expressivity for efficiency); some applications may accept non-standard rules of inference specialized to some particular application class or type, and some users may prefer rules that seem "illogical" to other users (such as "I will assume that anything my mother says is true, is true"). To be able to accommodate the wide range of users and applications, the policy-aware Web will need to support and be able to tolerate many kinds of different "proofs" being used for many different purposes.
4. Inconsistency must be handled in some way that does not cause the downfall of the Web. In many rule- and proof-based systems, anything can be derived from an inconsistency; thus these systems are generally

defined in a way that no inconsistency can be tolerated. On an open system like the Web, inconsistency is inevitable, and the policy-aware Web must have means to deal with it. This is particularly mandated for privacy and security applications where it can be assumed that some users will try to "raid" information sources. If it was possible to defeat the policy-aware Web by simply asserting "X" at one point and "not X" at another, the system would certainly not survive long in any useful state. Past work has defined "paraconsistent" logics that handle inconsistency in logic programming languages and deductive databases. A similar approach to handling inconsistency will be needed for the Web.

All of the capabilities above have been explored as separate research topics in a number of fields. However, to date no end-to-end approach that can combine all of the above has been developed. In the remainder of this chapter, we describe the steps we have taken to provide these capabilities and present an example of how the mechanisms we describe can be realized using tools that we have developed. Future work focuses primarily on the issue of dealing with inconsistency with the scaling of these tools to work on the Web and the development of a prototype environment (controlling access to personal photographs) that we are building to explore these issues.

Rules Engines as a Foundation of the Policy-Aware Web

Recalling the layer-cake diagram from the beginning of this chapter, the Semantic Web and the functionality we want requires a "stack" of standard languages to be designed for facilitating the interoperability of tools—just as the Web itself required the definition of a markup standard (HTML), so too does the Semantic Web stack necessitate shared languages. More importantly, by building on already existing Web standards, policy awareness will not require changing the basic architecture of the Web—after all, our goal is for policy awareness to eventually be built right into the user's Web client (and displayed through their Web browser).

For a rule language to meet our needs, it will have to be realizable in a form where the rules can be published, searched, browsed, and shared using the well-known HyperText Transfer Protocol (HTTP 1.1). The rule language must therefore be defined in a way that can take advantage of the Web protocols enabled by being realized in XML documents and exploiting the document tagging properties thereof, using the linking capabilities provided by the Resource Description Framework (RDF), the class definitions enabled by RDF Schema, and the more powerful ontological agreements enabled by the Web Ontology Language OWL.[4]

Currently, designing a rule language that is syntactically realized in XML and, preferably, compatible with RDF is an active research area. Current approaches include a proposal for RuleML; an XML-based rule standard (Boley, Tabet & Wagner, 2001); a recent proposal called SWRL, which builds the rules on top of OWL (Horrocks et al., 2003); and a very powerful logic language, called the SCL (Standard Common Logic, Menzel & Hayes, 2003) that is intended as a Web-based successor to the earlier KIF language. Given our concern for transparency, it is clear that a human-readable form of the language is important; RDF/XML format is often overly verbose and difficult for humans to interact with. Therefore, we base our work on "Notation 3" (or N3 as it is more commonly known), which was designed by Berners-Lee (2000) and is now actively supported by a growing open-source development community. N3 is an RDF-based rule language that was designed based to be consistent with a number of Web Architecture principles. N3 is also designed to work closely with Cwm, an RDF-based reasoner, specifically defined for Web use, which we discuss in the next section.

Providing the details of the N3 rule language is beyond the scope of this chapter, but a simple example should suffice to show a couple of the special features of the language. The simple N3 rule

```
{?x cs:teaches ?y. ?y cs:courseNumber math:greaterThan 500 } =>{ ?x a cs:professor}
```

states that if *X* teaches *Y,* where *Y*'s course number is greater than 500, then *X* must be of type professor. Note that qnames are used (and would be defined elsewhere in the document) to denote the unique URIs of each of the entities in the formula. The special qname "log" is used to denote logical properties while, in this case, the qname "math" is being used to invoke mathematical functions. This rule can be rendered into RDF/XML in an automated way, and in that form, although ungainly, the N3 becomes a valid XML document (the importance of which we will return to).

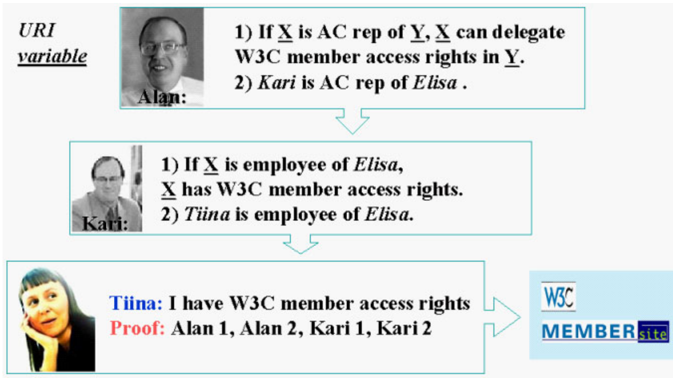
The N3 rules language has co-evolved with the design of a reasoner which can process the rules and evaluate them appropriately in a Web context—that is, a Web-based prover must be able to handle those procedural attachments crucial for working on the Web and must have a means for reasoning about a set of assertions that can be accessed on the Web using standard Web protocols. Cwm (Berners-Lee, 2000) is a reasoner that has been specifically developed to work in the Web environment. Cwm is a forward-chaining reasoner that can be used for querying, checking, transforming, and filtering information on the Web. Its core language is RDF, extended to include N3 rules. One of the key features of Cwm is its ability to include a number of specialized modules, known as built-ins, which allow a number of different functions to be evaluated during rule processing. The specific procedures are those needed for processing information on the Semantic Web, ranging from simple functions like *math:greaterThan* which invokes a mathematical function to *log:semantics* which allows information to be fetched from the Web and parsed or *crypto:verify* which verifies a digital signature. Indeed, in Cwm the integration of the Web and inferencing goes even further: the inference engine can look up symbols on the Web to discover information which may directly or indirectly help to solve the problem in question. Predicates can be looked up to find OWL ontologies or queried so as to find the specific properties of individuals and classes defined elsewhere on the Web.[5]

Cwm's Web-specific built-ins, which are integrated into its inferencing algorithms, make it a useful tool and will serve as the primary tool used for checking rules, handling certificates, generating and checking proofs, and controlling access. Cwm has been used for prototyping the capabilities discussed in this chapter. (Current work is exploring how to scale Cwm. Approaches include the development of a new RETE-based algorithm for Cwm and an analysis as to whether it is possible to use deductive database techniques to improve Cwm's performance. In particular, we are exploring whether a recent approach to magic sets (Behrend, 2003) can be used to provide database like scalability to Cwm under certain circumstances, despite it being more expressive than Datalog.)

Implementing Rule-Based Access with a Semantic-Web Proof Engine

Cwm allows us to implement rule-based access control on the Web in several ways. First, Cwm is able to check whether an access request can be granted in the "base case" where either a signed certificate or a grounded assertion is

Figure 2. Example of a proof-based access to a Web



presented. (A grounded assertion is one where a URI is used to point to an assertion that can be checked on the Web using HTTP-Get.) In these cases, Cwm checks that the antecedent of a policy rule indeed matches the subject's access (similar to the approach used in rule-based MAC). In more complex cases, Cwm can check a set of such assertions to make sure they are all valid and then check that they form a "proof", showing the rule or rules for access have been met.

Consider the example shown in Figure 2 (which is a more complicated version of the Web file-access rule shown previously). In this case, access to some files on the W3C Web site will be granted to a user if that user can prove they work for a member of the W3C. Further, the W3C can delegate the certification of users to an individual at a member organization. In this example, when Tiina requests access, she can prove that she meets these rules by showing that Alan had the right to delegate the authority, that Alan delegated the authority to Kari, and that Kari certified that Tiina is an employee of his organization. Given the rules shown and the grounded assertions (i.e., the rules with the variables replaced by the instance data), Cwm is able to demonstrate that the assertion that Tiina has access to is consistent with the policy and can grant her access.

A working version of this example is part of an online Cwm tutorial which can be found at http://www.w3.org/2000/10/swap/doc/Trust. The example combines rule-based reasoning with the use of built-ins for cryptographic access to prove access should be given. The key rule in the system is:

```
this log:forAll :d, :k, :k2,.
{    :request a acc:GoodRequest } is log:implies of
{
          :request acc:forDocument :d;
               acc:requestSupportedBy  :k.

 []           acc:certSupportedBy :k2,
                      log:includes { :k a acc:RequestKey }.
 []           acc:certSupportedBy [a acc:MasterKey];
                      log:includes { :k2 a acc:MemberKey }.
```

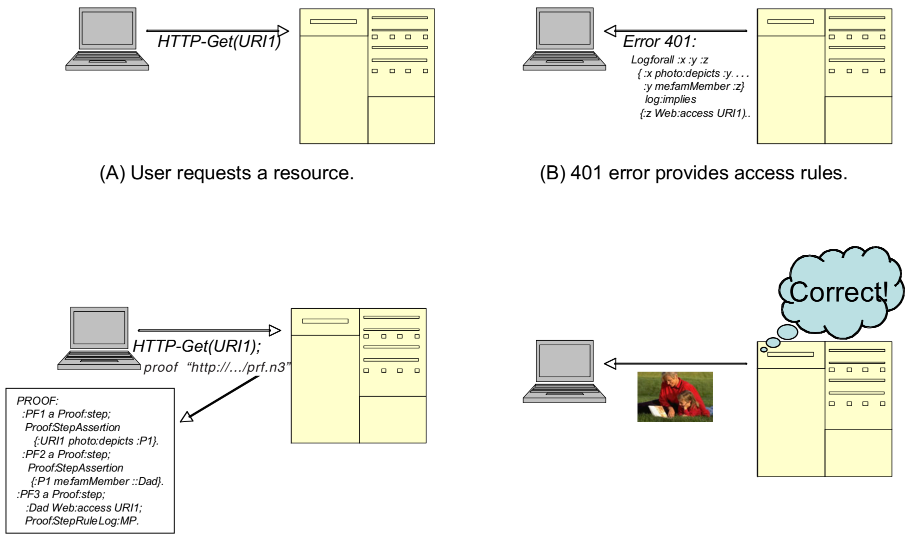
This rule states: if a request is supported by a key, and there is a certificate—signed itself with k2—which says k is a good request key and that there is some other certificate, signed with the master key, that says k2 is a member key, then the request is a good request.

A problem with our current use of Cwm in this example is that although it correctly meets the rules stated in Figure 2, it requires the bulk of the reasoning to be done by Cwm on the server's side. Thus, while the rules as to who can generate what certificates is somewhat distributed, the proof as to the trustworthiness of the certificates is generated by Cwm on the site accessed. We are exploring a system that is both more scalable and distributed by using Cwm to generate proofs on the client side and then to transmit these (via http) to the server, which then only has to check the proof to see if it is both grounded and consistent.

The protocol for doing this is quite straightforward and uses the standard Web protocol. A user attempts to access a site as usual, clicking on a Web link. This initiates an HTTP-GET request to the URI in question. Assuming the requested URI is protected by a set of access rules, the user will receive a 401 "Unauthorized" response. 401 errors, defined in IETF RFC 2617 (http://www.ietf.org/rfc/rfc2617.txt), are extensible by the addition of new tokens that specifically define the authentication schemes, and we take advantage of this. We return, as a part of the 401 response, the N3 description of the access rules. The client can then generate a proof and transmit the URI for that proof to the server as part of a follow-up HTTP-GET requesting authorization. Figure 3 illustrates this use of the 401 protocol for rule and proof exchange.

Although the protocol is straightforward, the transmittal of proofs requires its

Figure 3.



(A) User requests a resource.

(B) 401 error provides access rules.

(C) Proof is generated and pointer is sent in new HTTP-Get request.

(D) Proof is checked, and confirmed, and the transaction succeeds.

own syntax and semantics. Several research groups (cf. Pinhiero da Silva, McGuinness, & McCool, 2003; Hendler, 2004) have been working on developing languages for the exchange of proofs on top of the OWL language (essentially, simple proof ontologies). These ontologies are relatively straightforward and allow the proof to be represented as a set of steps, each containing a list of previous steps they depend on and the rationale used to produce the new clause. Thus, for example, a step in a proof might look like (in N3):

```
:PF8 a Proof:Step;
  Proof:StepAssertion
    {:Auth :Kari :URI1};
  Proof:StepRule log:ModusPonens;
 Proof:StepDependsOn (:PF6 :PF7).
```

stating that the assertion containing the fact that Kari authorized access to some URI followed by the logical rule modus ponens from two previous steps of the proof.

One of the more interesting aspect of proof checking on the Web is the proofs presented may contain not just traditional logics, but also extended (higher-order) logics or even proof steps grounded in "non-logical" justifications. One of the most important examples of the use of "non-standard" logics on the Web is Proof-Carrying authentication (PCA, Appel & Felten, 1999; Bauer, 2003). The Princeton team working on PCA has designed and implemented a general and powerful distributed authentication framework based on a higher-order, constructive logic and postulated that higher-order logics can be used as a bridge between security logics in a way that would enable authentication frameworks based on different logics to interact and share resources. We believe this is an important technology for the Policy-Aware Web, and we are working on extending our proof language (and Cwm's processing thereof) with a "quasi-quoting" facility to handle higher-order constructs such as those used in PCA in the open and distributed framework we are advocating.

In many cases, "proof steps" will actually be justifications that must be shared between the parties without the ability to appeal to a formal model theory or other proof of logical correctness. Thus, on the Web, steps in a proof may be made by reference to an agreed upon "oracle" rather than to a logical mechanism. For example, suppose we have a rule that says you can only have access to an entire passenger roster if you can authenticate that you work for a Federal Agency and you can produce the name of at least one passenger who has purchased a ticket for the flight. The former can be validated by the sort of keys and authentication discussed above, but the validation that passenger you have named has actually purchased a ticket may require that a separate airline system check its purchase database and make the answer available on the Web. One of the steps in the proof is to essentially say something like "you can find this asserted at the URI

http://owl.mindswap.org/people/pages/pages.py?person=%7B%27link%27%3A+%27http%3A%2F%2Fowl.mindswap.org%2F2003%2Font%2Fowlweb.rdf%23JimHendler%27%7D

which is clearly a non-standard logical mechanism. However, if the system checking the proof agrees that that Web page is on a trusted server, and the assertion can be found there, then this can be a valid (and important) justification. Different users, of course, may have trust in different servers, may be willing to accept different sets of axioms in the proofs conveyed, and so forth.

Our policy-aware approach to access control is a response, in part, to the observation that typical security architectures involve the requesting party doing very little computation—typically, just providing a username/password or perhaps computing a message digest and/or digital signature—and the party providing and controlling access being obliged and trusted to derive a justification based on the request credentials and some access control policy data (e.g., the file permission bits). Execution of even the relatively inflexible policies described above depends on enormous trusted computing bases. At W3C, the trusted computing base starts with the entire linux and solaris kernels, apache, php, mysql; and we are constantly developing custom Web-based tools; a bug in any of them puts our access control at risk. The decision of how much software to trust can go beyond the boundaries of our organization: if W3C wants to prove to the satisfaction of some outside party that our policies have not been violated, we would need to audit this entire computing base to the outside party's satisfaction.

If we shift the burden of deriving the access justification to the requesting party, who transmits that justification to the controlling party, who need only check it, the resulting system (a) has a much smaller trusted computing base (only the part that verifies justifications) and (b) is much more transparent: any third party can audit that the justification is valid.

We contend that such "social" proof mechanisms will be a critical part of the Web access mechanism, and we must handle them. Our work on conveying proofs therefore needs to be able to do more than say that "X is true". Rather, we must represent that "X is asserted to be True at Location Y" (or possibly a set of locations as in the Tiina example above). SHOE, an early Web Ontology Language developed at the University of Maryland, used a "claims logic" (Heflin, Hendler, & Luke, 1998; Heflin, 2001) to differentiate between a statement found on the Web and the resource asserting it. (The OWL language, the current Web ontology standard, chose to use a more traditional model theoretic approach (based on description logics) rather than use a less-standard claims logic). However, several features of the claims logic turn out to be powerful for proof checking on the Web, and we are revisiting these in our work. In particular, there appear to be three kinds of "proof steps" that are atypical in the standard proof checking literature:

- In the case we have been calling "grounded" above, the truth value of an assertion is checked by a "sensing action" (an HTTP-GET or checking a certificate) on the Web.

- The second case is where one proof checker may invoke another—for example, a specialized reasoning component may choose to delegate a complex piece of a proof to a more general system to check whether some rule holds.
- The third case is where some sort of "meta reasoning" is needed about the assumptions. For example, if "Site X claims P" and if I believe that Site X is trustworthy, then I am willing to believe that P is true (even though there is no logical theory backing it up).

As we continue our development of a Cwm-based proof checking tool that can handle the protocol shown earlier (Figure 3), we are exploring how best to handle these cases.

Future Work

There are several key challenges to extending this work and making it practical for Web deployment. First, coherent representation and operation with inconsistency inherent in an open system such as the Web remains an unsolved problem. Most logic-based systems built to date are very intolerant of inconsistency. Most research has therefore focused on removing inconsistency by limiting expressivity, controlling data entry to disallow entries that could cause inconsistency, and/or by strongly enforcing integrity constraints and other similar mechanisms. Second, while we have demonstrated the design and implementation of one Policy-Aware application (W3C site access control), there still remains substantial work to do in developing protocols and user interface strategies to enable the full range of transparency, compliance management, and accountability required for the Policy-Aware Web.

Inconsistency

Unfortunately, in an open system such as the Semantic Web, it is inevitable that inconsistency will occur, and we contend that none of these approaches is likely to work: for our system to be powerful enough to control access to non-structured documents, Web services, and multimedia, it must have at least the expressivity of OWL (which allows inconsistency), we cannot control data entry in an open system, and integrity constraints are difficult to maintain, let alone enforce, in a distributed and extensible system. Social mechanisms for enforcing consistency are also likely to fail, as inconsistency may be the result of error (i.e., putting data in the wrong field on a form), serious disagreement (i.e., the Web sites of abortion supporters and opponents would be unlikely to have consistent ontologies), or maliciousness (i.e., the deliberate introduction of inconsistency to attempt to circumvent the very policies we are trying to enforce). Thus, developing an approach where inconsistency can be tolerated and kept from causing harm is one of the key areas of research in our work.

The primary problem with inconsistency is that in classical logics, not only are inconsistent statements false, but they entail every other statement whether related or not. Thus, the mere presence of an inconsistency in such systems renders everything meaningless. Rooting out inconsistency becomes essential as nothing useful can be done once the knowledge base becomes inconsistent. Paraconsistent logics are logics that tolerate inconsistency by blocking the inference from a contradiction to arbitrary conclusions. In essence, these logics are constructed so that the effects of contradictions are localized and do not propagate. Thus, if X & -X are asserted, it will not cause a system to believe Y, Z, Q, and so forth unless these are specifically affected by the contradiction. Different paraconsistent logics localize contradictions in different ways: non-adjunctive logics (da Costa & Dubikajtis, 1977; Schotch & Jennings, 1980) prevent contradictory assertions from automatically forming self-contradictions (i.e., the truth of X and the truth of Y does not necessarily imply the truth of X AND Y); relevance logics (Routley et al., 1982; Restall, 1993) prevent explicit self-contradictions from entailing conclusions that are not directly related to the contradiction; and multivalued paraconsistent logics (Asenjo, 1966; Dunn, 1976) permit assertions to have truth values other than 1 or 0. Although all of these have been explored in the literature, few examples of paraconsistent reasoners have been implemented.

One notable exception is in the area of annotated logics for logic programming languages and especially the work of Kifer and Subrahmanian (1989) and Subrahmanian (1994). Annotated logics are an effective paraconsistent formalism for a number of reasons: they have clear semantics and a proof theory; they are a clean extension of FOL; and they are reasonably intuitive to work with. From a Semantic Web viewpoint, they are also desirable as annotated logics fit well with the Semantic Web's focus on "triples"—the natural locus for annotations (in fact, the claims logic of SHOE, described above, was implemented as an annotation framework in XSB). One difficulty with bringing annotation logics to the Web is in determining what set of annotations (and logic) offers the right balance of user transparency, scalability, and expressivity. While annotated logics allow the non-destructive presence of inconsistency, they often offer many incompatible ways of localizing the inconsistency, and the effects of these on security policies have not been carefully explored. Annotated logics also have tended to work in a centralized and controlled framework, so integrating them into an open and multi-perspective framework like the Web produces a number of challenges. We are exploring how to develop an instantiation of the Kifer and Subrahmanian framework that is implementable in Cwm so that we can test various different annotation theories for their efficacy and usability.

Transparency, Compliance, and Accountability Revisited

The technical approach described in this chapter has focused on the use of, and extensions to, N3 and Cwm for use in rule-based access on the Web. However, our goal of creating a policy-aware infrastructure for the Web includes more than just these basic infrastructure components. Achieving the triple goal of transparency, compliance management, and accountability requires exploration of the process of developing and agreeing on policy vocabularies, and addressing a variety of complex user interface challenges in order to represent policy aware information to the general user. By exploring application models to enable communities to take advantage of policy description, we believe we will be able to extend the reach of Semantic Web tools to meet policy-aware requirements.

Policy awareness begins with *transparent* access to rules associated with any given resource. While we have shown that it is possible to put rules on the Web and to use HTTP and RDF infrastructures to exploit them, we are still far from the full realization of policy awareness as described in the Being Policy Aware section. Making the rules explicit, publishable, and exchangeable via HTTP provides a significant improvement in transparency, and from a programmer's point of view meets our stated goals. However, putting this capability into the hands of end users will require much more work to determine how to build user interfaces (Ackerman, Darrell, & Weitzner, 2001) that provide usable access to social rules and to tools that communities can use to decide on and develop rules.

The access control mechanism described in the Rule-Based Access and the World Wide Web section illustrates a simple case of the *compliance manage-*

ment, the second important attribute of the Policy-Aware Web. While this access control mechanism demonstrates that rules engines can be used on the Web to mediate access, there is much more to be done to enable full policy awareness. Consider the example of a set of people exchanging photographs on the Web. A person posting a photo to a site might wish to know what the site's policy is with respect to sharing the photos. Similarly, a user wishing to share personal information might wish to publish a set of photos but control who can see them. Publishing a picture and saying "my friends can see it" seems simple but actually raises complex issues. This is because we expect rules to be evaluated in a multi-party, multi-transaction social setting.

Social rules require careful consideration of unanticipated "transitive closure" when applied in more sophisticated, but likely more typical, community applications. Consider the case where the user took a potentially embarrassing photo (say the unlikely case of a picture of someone drinking too much at the WWW conference). Publishing this to friends seemed straightforward to the user, but he forgot that some of his friends also worked at his company.

One of these people saw the photo and republished it to his "business associates," which violated the original intent of the photographer. Further, the photographer (whose identity is encapsulated in the EXIF information in the photo) is unable to demonstrate that he was not the one who shared the photograph in the first place, earning the enmity of the photo's subject and other such social detriment. In this case, building the third component of policy awareness, *accountability*, would help the community of photo sharers to figure out how and even why a photo got shared beyond the intended constraints. Perhaps someone made a mistake, or perhaps someone played a malicious joke. Accountability mechanisms that reconstruct the proofs presented to gain access and establish what policy statements were associated with the image when it appears outside the boundaries established by the community can help to establish whether the act was intentional or inadvertent.

Conclusion

The infrastructure discussed in this chapter is a starting place for exploring this important problem and for allowing the greater sharing of personal information. However, it is just a start, and much work remains to be done if we are to eventually see a truly Policy-Aware Web. We have described the development of a rule-based policy management system that can be deployed in the open and distributed milieu of the World Wide Web. Combining a Semantic Web rules language (N3) with a theorem prover designed for the Web (Cwm), we have shown that it is possible to apply rules on the Web using the Hypertext Transport Protocol (HTTP) to provide a mechanism for the exchange of rules and, eventually, proofs. We have also shown how this mechanism can provide a base for a Policy-Aware infrastructure for the Web and have argued for the necessity of such an infrastructure.

We anticipate that policy awareness tools will enable the Semantic Web to address a wide range of policy requirements: questions such as who owns the copyright to a given piece of information, what privacy rules apply to an exchange of personal information, and what licensing terms apply to a particular piece of genetic information are all examples of social needs that policy awareness can help mediate. In testimony to the United States Congress in 2000, Daniel Weitzner argued:

This same interactivity, the bi-directional ability to exchange information from any point to any other point on the Net has brought about significant threats to individual privacy. For the same communications mechanisms that give individuals the power to publish and access information can also be used, sometimes without the user's knowledge or agreement, to collect sensitive personal information about the user…. Our goal is to use the power of the Web, and enhance it where necessary with new technology, to give users and site operators tools to enable better knowledge of privacy practices and control over personal information. (Daniel J. Weitzner, Testimony to US Senate Commerce Committee, May 2000)

Policy awareness is not alone sufficient to solve the pressing public policy problems raised by the interaction of the Web and society, but we believe that Policy-Aware infrastructure is a necessary part of enabling human institutions and communities to adapt to this new environment.

Acknowledgments

The authors thank Joe Pato, Hewlett-Packard Laboratories, and Poorvi Vora, George Washington University for insightful discussions of trust in ad hoc online communities, and Ted Wilson of Hewlett-Packard for initial exploration of policy capable infrastructures. Some or all of the authors were supported in part by grants from the DARPA Agent Markup Language program, DARPA Fast C2 Applications project, and funding from the Army Research Laboratory, the National Institute for Standards and Technology, and other Department of Defense agencies as well as a grant from the National Science Foundation which directly supports this work. We also thank Fujitsu Laboratory of America College Park, NTT Corp., and Lockheed Martin for funding of the University of Maryland MIND Laboratory.

References

Ackerman, M., Darrell, T., & Weitzner, D.J. (2001). Privacy in context. *Human-Computer Interaction, 16*, 167-176

Appel, A., & Felten, E. (1999). Proof-carrying authentication. *Proceedings of the 6th ACM Conference on Computer and Communications Security,* November, Singapore.

Asenjo, F.G. (1966). A calculus of antinomies. *Notre Dame Journal of Formal Logic, XVI*, 103-5.

Barbour, G. (2002). *Program modeling: A machine learning approach to intrusion detection.* Unpublished doctoral thesis, University of Maryland, Computer Science Department.

Barbuti, R., Bernardeschi, C., & De Francesco, N. (2004). Analyzing information flow properties in assembly code by abstract interpretation. *The Computer Journal, 47*(1), 25-45

Bauer, L. (2003). *Access control for the Web via proof-carrying authorization.* Unpublished doctoral thesis, Princeton University, Princeton, NJ.

Bauer, L., Schneider, M., & Felten, E. (2002). A general and flexible access-control system for the Web. *Proceedings of the 11th USENIX Security Symposium,* August, San Francisco.

Behrend, A. (2003). Soft stratification for magic set based query evaluation in deductive databases. *Proceedings of the International Conference on Management of Data and Symposium on Principles of Database,* San Diego.

Berman, J., & Weitzner, D. (1995). *Abundance and user control: Renewing the democratic heart of the First Amendment in the age of interactive media.* Technical Report No. 104 Yale L.J. 1619), New Haven, CT: Yale University.

Berners-Lee, T. (2000). Primer: Getting into RDF & Semantic Web using N3. Retrieved April 30, 2005, from *http://www.w3.org/2000/10/swap/Primer*

Berners-Lee, T. (2000, December). *Semantic Web.* Presentation at XML 2000, Washington, DC. Retrieved April 30, 2005, from *http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html*

Berners-Lee, T. (2000). CWM—A general purpose data processor for the Semantic Web. Retrieved April 30, 2005, from *http://www.w3.org/2000/10/swap/doc/cwm.html*

Berners-Lee, T., Fielding, R., & Frystyk, H. (1996, February). Hypertext Transfer Protocol - HTTP/1.0. HTTP Working Group.

Berners-Lee, T., Hendler, J., & Lassila, O. (2001, May). The Semantic Web: When the Internet gets smart. *Scientific American*.

Boley, H., Tabet, S., & Wagner, G. (2001). Design rationale of RuleML: A markup language for Semantic Web rules. *Proceedings of SWWS'01*, July/August, Stanford, CA.

Boneh, D. & Franklin, M. (2001). Identity based encryption from the Weil pairing. *Crypto 2001, LNCS 2139*, 213-229.

Burrows, M., Abadi, M., & Needham, R. (1989). A logic of authentication. *Proceedings of the Royal Society of London, LNCS 426* (pp. 233-271).

da Costa, N.C.A., & Dubikajtis, L. (1977). On Jaskowski's discussive logic. In A.I. Arruda, N.C.A. da Costa, & R. Chuaqui (Eds.), Non-classical logics, modal theory and computability (pp. 37-56). Amsterdam, The Netherlands: North-Holland Publishing Company.

Didriksen, T. (1997). Rule based database access control: A practical approach. *ACM Workshop on Role-Based Access Control,* 143-151.

Dunn, J.M. (1976). Intuitive semantics for first degree entailment and coupled trees. *Philosophical Studies, XXIX*, 149-68.

eXtensible Access Control Markup Language, OASIS Technical Committee. Retrieved April 30, 2005, from *http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml*

Ferraiolo, D., Kuhn, D., & Chandramouli, R. (2003). *Role-based access control.* Norwood, MA: Artech House.

Heflin, J. (2001). *Towards the Semantic Web: Knowledge representation in a dynamic distributed environment.* Unpublished doctoral thesis, University of Maryland, College Park.

Heflin, J., Hendler, J., & Luke, S. (1988). Reading between the lines: Using SHOE to discover implicit knowledge from the Web. *Proceedings of the AAAI-98 Workshop on AI and Information Integration,* Madison, WI.

Hendler, J. (2004). *An OWL full ontology for creating proof instances on the World Wide Web.* Manuscript submitted for publication.

Horrocks, I., Patel-Schneider, P., Boley, H., Tabet, S., Macgregor, R., Grosof, B., & Dean, M. (2003). SWRL: A Semantic Web rule language combining OWL and RuleML, version 0.5. Retrieved April 30, 2005, from *http://www.daml.org/rules/proposal/*

Kagal, L., Paoucci, M., Srinivasan, N., Denker, G., Finin, T., & Sycara, K. (2004, July). Authorization and privacy for Semantic Web services. *IEEE Intelligent Systems (Special Issue on Semantic Web Services)*.

Kifer, M., & Subrahmanian, V.S. (1989). On the expressive power of annotated logic programs. In E.L. Lusk, & R.A. Overbeek (Eds.), *Proceedings of the North American Conference on Logic Programming* (pp. 1069-1089).

Knowledge Interchange Format. Retrieved April 30, 2005, from *http://logic.stanford.edu/kif/kif.html*

Kyte, T. (2000, August). Fine grained access control (aka DBMS_RLS). Retrieved April 30, 2005, from *http://govt.oracle.com/~tkyte/article2/index.html*

Menzel, C., & Hayes, P. (2003). Standard common logic. Retrieved April 30, 2005, from *http://ceur-ws.org/Vol-82/SI_paper_12.pdf*

Ott, A. (2001). The rule set based access control (RSBAC) Linux kernel security extension. *Proceedings of the International Linux Congress,* November, Enschede, The Netherlands.

OWL Web Ontology Language Semantics and Abstract Syntax, W3C Recommendation. Retrieved April 30, 2005, from *http://www.w3.org/TR/owl-semantics/*

Pandey, R., & Hashli, B. (1999). Providing fine-grained access control for Java programs. *Concurrency: Practice and Experience*.

Pinheiro da Silva, P., McGuinness, D., & McCool, R. (2003). Knowledge provenance infrastructure. *IEEE Data Engineering Bulletin, 26*(4), 26-32.

The Platform for Privacy Preferences 1.1 (P3P1.1) Specification, W3C Working Draft. Retrieved April 30, 2005, from *http://www.w3.org/TR/P3P11/*

PORTIA: Privacy, Obligations, and Rights in Technologies of Information Assessment. Retrieved April 30, 2005, from *http://crypto.stanford.edu/portia/* — see also: Dan Boneh, Joan Feigenbaum, Avi Silberschatz, and Rebecca Wright, to appear in the *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 2004.

RDF Semantics, W3C Recommendation. Retrieved April 30, 2005, from *http://www.w3.org/TR/rdf-mt/*

Reno vs. ACLU, 521 US 844 (1997). Retrieved April 30, 2005, from *http://supct.law.cornell.edu/supct/html/96-511.ZO.html*

Restall, G. (1993). Simplified semantics for relevant logics (and some of their rivals). *Journal of Philosophical Logic, XXII*, 481-511.

Routley, R., Plumwood, V., Meyer, R.K., & Brady, R.T. (1982). *Relevant logics and their rivals.* Atascadero, CA: Ridgeview.

The Rule ML Initiative. Retrieved April 30, 2005, from *http://www.ruleml.org/.* See also [BoTW01].

Sandhu, R., Coyne, E., Feinstein, H., & Youman, C. (1996). Role-based access control models. *IEEE Computer, 29*(2), 38-47.

Schotch, P.K., & Jennings, R.E. (1980). Inference and necessity. *Journal of Philosophical Logic, IX*, 327-340.

Schunter, M. (Ed.). (n.d.). The enterprise privacy authorization language. IBM Corp. Retrieved April 30, 2005, from *http://www.zurich.ibm.com/security/enterprise-privacy/epal/Specification/index.html*

The Security Assertion Markup Langauge, OASIS Technical Committee. Retrieved April 30, 2005, from *http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security*

Shamir, A. (1985). Identity-based cryptosystems and signature schemes. *Advances in Cryptology—CRYPTO '84, LNCS 196*, 47-53.

Simple Distributed Security Infrastructure. Retrieved April 30, 2005, from *http://theory.lcs.mit.edu/~cis/sdsi.html*

Simple Public Key Infrastructure (open-source project). Retrieved April 30, 2005, from *http://sourceforge.net/projects/spki/*

Subrahmanian, V.S. (1994). Amalgamating knowledge bases. *Transactions on Database Systems, 19*(2), 291-331.

Swartz, A., & Hendler, J. (2001). The Semantic Web: A network of content for the Digital City. *Proceedings of the 2nd Annual Digital Cities Workshop,* October, Kyoto, Japan.

Weitzner, D. (2000, May). Testimony before the United States Senate Commerce Committee Hearing on Online Privacy.

Weitzner, D. (2004). The transparency paradox: Privacy design strategies for open information networks [Extended Abstract]. Retrieved April 30, 2005, from *http://www.w3.org/2004/05/loc-priv-transparency-extab.html*

Endnotes

1 Actually, a newer version of the W3C access system is built on the rule-based approach and uses some of the technology we describe later in this chapter (see http://www.w3.org/2001/04/20-ACLs for details).

2 Most XML-based security work, for example, focuses on tagging individual parts of a document with a list of applicable roles and the use of XML signatures for checking whether an incoming request matches those roles. Although this approaches the issue of finer-grained access, it does not address the other problems we discuss and primarily focuses on documents, not other Web resources.

3 Attribute certificates are generally used in PKI schemes for aspects of a transaction that have a shorter lifetime than the public-private key pair (cf. http://ospkibook.sourceforge.net/docs/OSPKI-2.4.6/OSPKI/pkix-concepts.htm).

4 The documents describing all of these languages can be found at http://www.w3.org—the W3C Web site.

5 CWM was originally an acronym for “Closed World Machine”, but it rapidly developed into a more powerful, open-world tool that is being used for the sort of distributed, open rule sets we describe in this proposal. Rather than renaming CWM, the development team dropped the acronym

and kept going. At the time of this writing, Cwm had about 100 built-ins. The list can be found at http://www.w3.org/2000/10/swap/doc/CwmBuiltins.html

Chapter II

Web Services Security

Carlos A. Gutiérrez García,
Sistemas Técnicos de Loterías del Estado, Spain

Eduardo Fernández-Medina Patón,
Universidad de Castilla-La Mancha, Spain

Mario Piattini Velthius,
Universidad de Castilla-La Mancha, Spain

Abstract

During the past few years, significant standardization work in the Web services (WS) technology area has been performed. As a consequence of these initial efforts, WS foundational stable specifications have already been delivered. Now, it is time for the industry to standardize and address the security issues that have emerged from this paradigm. Up until now, much activity has been carried out on this subject. In this chapter, we will specify the main security services that have to be addressed in the WS world as well as a description of the aspects already addressed and unaddressed. We will mention the main initiatives and their respective specifications that try to cover the distinct security issues within the WS outlook. Whenever possible, unaddressed security issues for each topic will be stated. In addition, current general security concerns will be detailed, and future research will be proposed.

Background

During the past few years, the Web services (WS) paradigm has achieved great popularity and is currently a new buzzword among middleware and enterprise software specialists. Technologies derived from this paradigm have reached such a maturity level that now they can be considered as the most promising full-fledged and standard integration solutions. Hence, WS has become a reality on which IT departments are basing their operations to achieve a direct alignment with the business operations they support (Casati, Shan, Dayal, & Shan, 2003). In fact, according to the most recent reports from IDC, over the next years, the market for WS solutions will grow steadily reaching $11 billion in 2008 (IDC, 2004).

Perhaps the most precise and practical definition of the term *WS* is provided by the W3C Web Services Architecture Working Group: "A Web service is a software application identified by a URI, whose interfaces and bindings are capable of being defined, described, and discovered as XML artifacts. A Web service supports direct interactions with other software agents using XML-based messages exchanged via Internet-based protocols" (W3C, 2004).

WS are distributed, decentralized, self-contained, self-describing; can be dynamically published, located, invoked; are language independent and interoperable, inherently open, standards based; and are able to be composed and provide well-defined services to certain service consumers (Endrei et al., 2004). Consequently, WS based-solutions must be concerned with typical security problems that are common to distributed communications, through a compromised channel, between two or more parties. Some of the major inherited security issues that WS technologies must address are authentication, authorization, confidentiality, data integrity, non-repudiation, and availability (Sedukhin, 2003). WS must address both the issues inherited from the distributed computing classical scheme and those arising from the new threats created by its own nature.

In addition, ways to protect service providers and service consumers are needed. For example, service providers can be protected by applying control access mechanisms to the services (OASIS, 2003b) or information (Shandu, Coyne, Feinstein, & Youman, 1996) they own or by guaranteeing non-repudiation of the interactions they perform. On the other hand, service consumers' protection is mainly focused on service trustworthiness and data privacy concerns. Service trustworthiness assures service consumers that the service they plan to use will act as expected (e.g., as indicated in its related meta-information). Data privacy deals with protecting sensitive and personal information collected by a service from unauthorized disclosures. We will analyze these concepts later on in this chapter.

Some of the new threats associated with WS technologies are enumerated below:

1. Risks appeared due to the publication on Internet of a complete and well-documented interface to back-office data and company's business logic. One of the main security problems associated with the adoption of WS is derived from the Internet publication of business interfaces through ports HTTP 80 or HTTPS 443 (O'Neill et al., 2003).
2. Protecting the Semantic Web by "ensuring that security is preserved at the semantic level" (Ferrari & Thuraisingham, 2004).
3. Context-aware and context-based protection at the document level (Bhatti, Bertino, Ghafoor, & Joshi, 2004; Bertino, Castano, & Ferrari, 2001). Documents usually have information with different "degrees of sensitivity" which require protection at different levels of security. Today, access control policies that govern access to the different security parts of the documents and an architecture enforcing these policies constitute a very important research area in the context of WS security.
4. Service trustworthiness. Dynamic discovery and composition of services implies that a Web service consumer may not know whether the services, individually or composed, are going to behave as expected. "Consequently, how to select **trustworthy** Web services remains a challenge" (Zhang, Zhang, & Chung, 2004).

And some of its main security objectives are:

1. Management of security policies in a large distributed WS environment (Chang, Chen, & Hsu, 2003).
2. Application-level, end-to-end, and just-one-context-security communications. Network topologies require end-to-end security to be maintained all across the intermediaries in the message's path. When data is received and forwarded on by an intermediary beyond the transport layer,

both the data integrity and any security information that flows with it may be lost (IBM & Microsoft, 2002).

3. Interoperability of the requirements and online security elements (WS-I, 2004).
4. Ability to federate the full information about the subjects enabling single sign-on environments and facilitating across-enterprise interoperability (Liberty Alliance Project, 2003).
5. Keeping sensitive users' attributes and identity private among trust domains.

Web Services Foundational Standards

WS may be considered a new computational paradigm that has been widely accepted as an implementation of SOA (Service-Oriented Architecture) (Erl, 2004; W3C, 2004). This fact has led the fundamental characteristics described by SOA to be the ones that have initially headed the major efforts in the industry standards development process. The SOA foundation is composed of basic services, their descriptions, and the basic operations (publication, discovery, binding, and invocation) that generate or use such descriptions.

The core WS specifications are XML, SOAP, WSDL, and UDDI. These broadly adopted industry specifications provide the foundational building blocks on which WS are designed and implemented. The bad news is that these four operative services specifications allow the creation of WS, but they do not state how to secure them. In fact, they contain in themselves security questions that must be answered:

- **XML and SOAP:** These specifications do not specify how to obtain integrity, confidentiality, authenticity, or reliability of the information they represent and transport.
- **UDDI and WSDL:** They should answer questions such as the following: Is the UDDI registry situated at a trustworthy location? How can we be sure that the published data have not been maliciously manipulated? Were the data really published by the business they are supposed to have been published by? Can we trust the business that published the services? Are

the services available at any moment? Can we trust the transactions that are produced from the execution of the business services? As we can notice from all these questions, an in-depth analysis of the security problems that an UDDI and WSDL architecture implies has to be carried out (Adams & Boeyen, 2002).

Web Services Security

Web Services Security Initiatives

At this point, the main WS standardization initiatives are the World Wide Web Consortium (W3C) and the Organization for the Advancement of Structured Information Standards (OASIS). Both consortiums are trying to standardize their vision, including security, of what WS should be and how it should contribute to the WWW world.

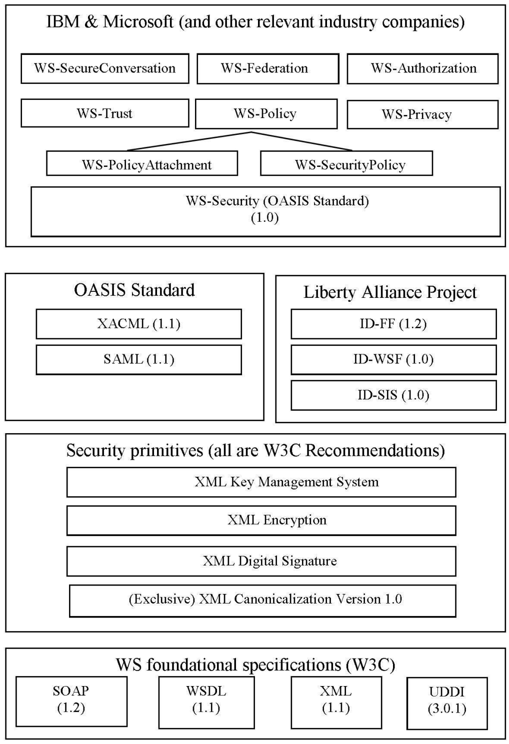
Another major organization in the WS world is the Web Services Interoperability Organization (WS-I). This open industry consortium has set, as its main objectives, managing the interoperability of WS across platforms, operating systems, and programming languages by means of standard protocols that allow the interoperable exchange of messages. Related to WS security, they released the initial version of the WS-I Basic Security Profile. Its main objective is to state what and how WS security specifications should be put to work in order to reach a full interoperability degree.

The Liberty Alliance Project, led by Sun Microsystems, has the purpose of defining a standard federation framework that enables services such as Single Sign-On. Thus, its major and initial intention is to define an authentication distributed system that allows intuitive and seamless business interactions to its clients.

Web Services Security Standards

In this section, we will describe the main standards being developed within each security area involved in securing WS. All these specifications are summarized in Figure 1.

Figure 1. Current security standards grouped by their standardization groups



They are grouped as follows:

- **Foundational specifications:** general purpose specifications which the rest of WS specifications are based on.
- **Security primitives:** standards that provide the XML low-level security primitives. It should be noted that XML Key Management System (Ford et al., 2001) is not a real XML security primitive, and it has been included as such just for classification reasons.
- **IBM & Microsoft (and others):** WS specifications created by both of them and other relevant industry companies. WS-Security was submitted to OASIS in order to carry out its standardization process.
- **OASIS:** security specifications developed by OASIS Technical Committees.

Table 1. Web services security standards grouped by the security topic they address

Authentication	***Simple message authentication***	WS-Security
	Digital signature-based challenge/response authentication	WS-Trust
	X509v3 certificate authentication	XKMS
	Single Sign-On	SAML authentication assertions, Liberty Alliance Project ID-FF , WS-Federation
Authorization	XACML XrML ODRL WS-Authorization	
Confidentiality	WS-Security + XML Encryption	
Integrity	WS-Security + XML Digital Signature	
Non-repudiation	MAY BE achieved by using XML Digital Signature + WS-Security Standard protocols not yet proposed	
Audit trail	Standard solution not yet proposed	
Trust management	WS-Trust XKMS SAML	
Security context establishment/ keys derivation	WS-SecureConversation	
Privacy	P3P, E-P3P, APPEL, EPAL, WS-Privacy	
Policy specification	WS-Policy + WS-SecurityPolicy XACML Web Services Profile	
Reliability	***Service***	WS-AtomicTransaction
	Transport	WS-ReliableMessaging, WS-Reliability, HTTP-R

- **Liberty Alliance Project:** It represents the group of specifications developed in the Liberty Alliance Project.

A major part of these specifications will be described next. Table 1 summarizes most of the standards and specifications that will be reviewed throughout this section. They have been classified in terms of the security issues they address.

Authentication

"When dealing with security, three are the main issues that need to be faced: authenticity, integrity, and confidentiality" (Ferrari & Thuraisingham, 2004). Authentication "is the degree to which the system verifies the identities of its externals before interacting with them" (Firesmith, 2004). One way a party can be authenticated is by providing a digital signature along with the message it sends. Then, the message's recipient can use the signature to verify the identity of the sending party. A standard way to create and verify XML-based digital signatures is specified by the W3C Recommendation XML Digital Signature (Bartel et al., 2002).

W3C XML Digital Signature is a W3C recommendation since 2002, accomplished from the joint work of W3C and IETF. It defines how to digitally sign XML content and how to represent the resulting information according to an XML schema. Digital signatures grant information integrity and may grant non-repudiation (e.g., an entity may not deny the authorship of a digitally signed bank transfer made through a WS).

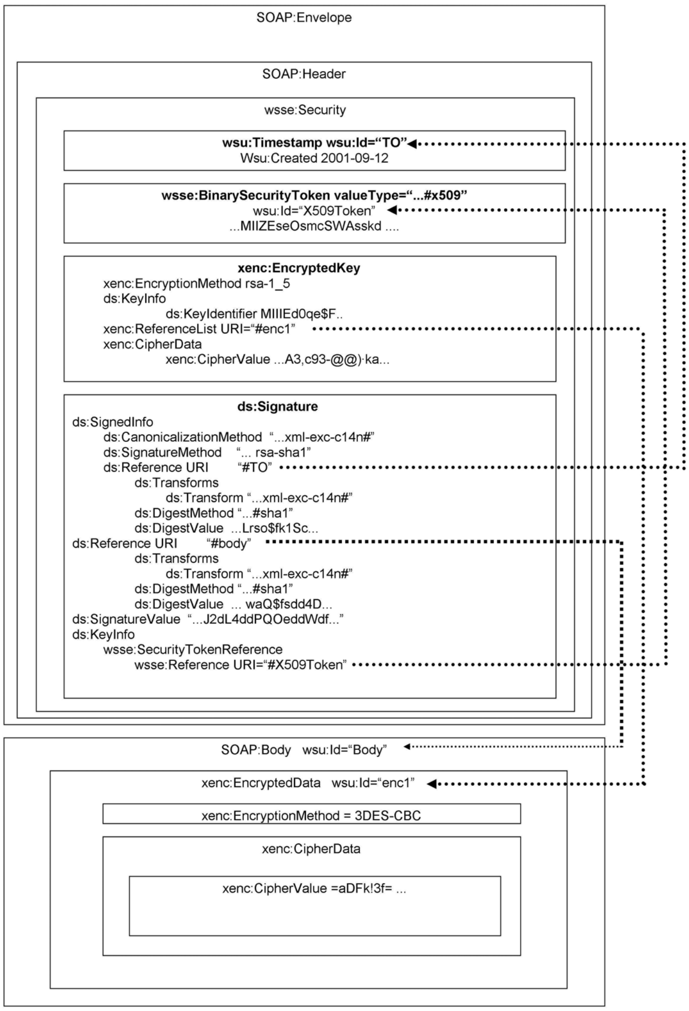
According to the XML Digital Signature specification, a digital signature can be applied to any kind of digital content, including XML. Signature creation and verification processes are defined by the specification. It is technology independent, so additional mechanisms are needed to define how it will be applied to WS message exchange.

In the context of messages exchanged among WS, these additional mechanisms are defined by the WS-Security specification (OASIS, 2004). As we have mentioned earlier in this chapter, IBM and Microsoft, among other IT companies, are jointly elaborating a series of specifications so as to compose an architecture, termed by Microsoft as Global XML WS Architecture*,* that will lead the development in the WS industry so that different products can interoperate within a secured context. These companies are the original authors of the WS-Security specification. IBM, Microsoft, and VeriSign developed and submitted it to OASIS, who is responsible for its standardization process. This is the specification which some additional specifications (some with publicized versions) that cover all aspects of security in WS have based their definition on. WS-Security, at the time of writing this chapter, exists in its 1.0 version, and it is placed as the base of the customized IBM and Microsoft security specification stack. Its purpose is to provide "Quality of Protection" to integration, adding the following properties to communication and messages: message integrity, confidentiality, and simple authentication of a message. WS-Security extends the SOAP messaging framework by defining a security header ("SOAP Module" extension in SOAP specification terms) that may include digital signatures (based on the W3C XML Digital Signature specification) and encrypted data (based on the W3C XML Encryption specification). Furthermore, it defines and explains the usage of collections of claims representing a user name/password or claims with binary content (e.g., x.509v3 certificates or Kerberos tickets) by means of the UsernameToken and BinarySecurityToken XML, respectively. These elements allow the transport of credentials for the authentication of the communication parties. By offering these properties, WS-

Security allows the easy incorporation of many existing distributed trust security models such as PKI and Kerberos.

WS-Security defines the new XML element *"Security"* that will act as the security SOAP header block. It is aimed at containing all the information related to the security that needs to be conveyed within the messages exchanged among the parties. For instance, we can suppose a scenario in which a WS A desires to send a message to a WS B. WS A knows (e.g., as it was indicated in the security policy attached to WS B) that it must send the message's payload encrypted with a symmetrical key K that needs to be sent within the message itself so that, subsequently, WS B can decrypt the message. WS A will encrypt

Figure 2. WS-Security example message



the symmetrical key with the public key contained within the X509v3 certificate of WS B. Figure 2 shows the structure of the message sent by WS A.

As we can see in Figure 2, in this case, the Security header block is composed of 4 main elements: the first one allows a timestamp to be embedded so that the freshness of the message's security context can be evaluated; the second one is a binary security token that is an X.509v3 certificate belonging to WS B (this certificate contains the public key used by WS A for encrypting the symmetrical key); the third one is an EncryptedKey element defined by the XML Encryption specification that represents the encrypted symmetrical key; finally, the Signature element is imported from the XML Digital Signature specification, and it represents the digital signature created over the payload and the timestamp of the message.

Data Integrity

This property guarantees that the information received by a WS remains the same as the information that was sent from the client. Integrity is addressed to by applying digital signatures over the data to be protected. As we have already stated in the previous section, the standard way to address this issue is by using the mechanisms specified by the XML Digital Signature and WS-Security standards.

Data Confidentiality

"Confidentiality means that your data is kept secret from unintended listeners. Usually, these listeners are eavesdropper" (Rescorla, 2001). Two main approaches are usually applied when looking for confidentiality: establishment of dedicated lines/Virtual Private Networks or, when data is sent over an untrusted network (e.g., Internet), use of encryption primitives at the application level. In WS security, one of the major objectives is to provide an end-to-end just-one security context so that the latter approach is used.

W3C XML Encryption (Imamura, Dillaway, & Simon, 2002) provides us with a model for encryption, decryption, and representation of XML document elements. W3C XML Encryption solves the problem of confidentiality of SOAP messages exchanged among WS nodes. It describes the structure and syntaxes of the XML elements that represent the encrypted information. In addition, it provides rules for encrypting/decrypting an XML document (or parts of it). The specification states that encrypted fragments of a document should be replaced by XML elements specifically defined in the recommendation. In order to recover the original information, a decryption process is also specified.

Non-Repudiation

In the WS world, it is necessary to be able to prove that a client used a service (requester non-repudiation) and that the service processed the client request (provider non-repudiation). This security issue is covered by means of public key cryptography, digital certificates, and digital signatures. XML Digital Signature W3C standard can be used to link a signature (over a message) to the identity of that message's sender so that the signer (sender) cannot deny signing the data (O'Neill et al., 2003). At this point, there are no standard Web services-based, non-repudiation protocols.

Authorization

Authorization is the degree to which access and usage privileges of authenticated externals are properly granted and enforced (Firesmith, 2004). In WS context, authorization is usually specified as the policies and mechanisms enforced by a WS provider in order to control access to the services it provides. The security access control frameworks that will be discussed in this section are eXtensible Access Control Mark-up Language, WS-Authorization, WS-Policy framework, XrML, and ODRL.

Today, the main WS authorization access control security standard is the XACML (OASIS, 2003b). XACML has been an OASIS specification since February 2003, and its main intention is to define an XML vocabulary for specifying the rules from which access control decisions can be enforced. XACML defines these access control rules depending on the requester characteristics, the communication protocol, and the authentication mechanism used. Moreover, XACML defines a request/response protocol that allows the realization of access requests to certain resource(s) (and the representation of their corresponding responses).

Currently, work on developing a particular XACML profile for Web services is being carried out so that it "enables XACML to be used for describing policies associated with Web service end-points and using them in a successful invocation" (Anderson, Proctor, & Godik, 2004).

Proceeding from the already mentioned IBM and Microsoft effort, two authorization and policy related specifications are proposed: WS-Authorization and the WS-Policy Framework (VeriSign et al., 2004). The not yet published WS-Authorization specification provides the mechanisms that facilitate the management of the information used for access policies and authorization. On the other hand, WS-Policy framework is formed by several specifications that make the definition of general purpose policies possible (being able to specify requirements, individual preferences or capabilities of the communicating parties) and security-specific policies. Moreover, it explains how to attach WS-Policy compliance policies to different WS related artifacts (WSDL interface definition, UDDI registry entries, etc.).

The XrML (eXtensible Rights Mark-up Language) is a security policy framework designed for specifying rights over digital resources within the context of Digital Rights Management workflows (Iannella, 2001). XrML "can identify the parties allowed to use those resources, the rights available to those parties, and the terms and conditions under which those rights may be exercised" (Iannella, 2001). XrML defines XML-based grant assertions according to four basic building blocks: the main one is to whom the grant is granted, the rights being granted, the resource on which rights are distributed, and the condition(s) to be addressed for the right to be exercised. A license is "the issuance of grant assertions by their issuing parties"; that is, a license is composed of a set of grants assertions, issuers' license identification information, and additional information such as a description of the license and a validity period. XrML supports WS "through the use and specification of XML, XPath, SOAP and UDDI". This support allows a WS to send rights-related information or handle rights-related operations.

The Open Digital Rights Language is also a language used for Digital Rights Management (DRM). It "supports an extensible language and vocabulary (data dictionary) for the expression of terms and conditions on any content including permissions, constraints, requirements, conditions, and offers and agreements with rights holders" (Iannella, 2002).

Trust Management

In WS, as a distributed computational paradigm, the need for negotiation, representation, and distribution of trust is essential (Grandison, Sloman, &

College, 2000) (Blaze, Feigenbaum & Lacy, 1996). Distributed software subjects face the problem of determining one another's trustworthiness which, today, is approached by requiring that the communicating parties have a previously established relationship (e.g., identity-based or capabilities-based).

In open systems (e.g., the Web) a new approach in which completely unfamiliar parties can establish a level of trust among them is needed. This new approach is based on users' credentials. A credential contains signed assertions that describe the owner's attributes, and its process consists of exchanging credentials iteratively to "build trust between the negotiation parties" (Skogsrud, Benatallah, & Casati, 2003). The main aspects that are being addressed by trust management solutions are:

- A way of representing credentials, for example, X.509v3 certificates in PKI infrastructures, PGP trust model (Zimmerman, 1994), non-standard and natural language-based credentials representation as in KeyNote (Blaze, 1999), or SAML in Trust-Serv (Skogsrud et al., 2003), and modern credentials types as role-based credentials as in Li, Mitchell, and Winsborough (2002).
- A way of representing trust policies (e.g., standard-based as PICS in REFEREE , non-standard solutions as XML control tables in Trust-Serv or XML policies in the IBM Trust Establishment Framework).
- A trust structure (e.g., centralized and hierarchical in PKI or plain and decentralized as in PGP).
- Policy lifecycle management: policy versioning impact on ongoing trust negotiation and policy migration handling (Skogsrud et al., 2003). "Systems change and evolve, so there is a need to monitor trust relationships to determine whether criteria which they are based on still apply" (Grandison et al., 2000).
- Logic-based formalisms that provide a formal trust representation.

Among the main specifications that tackle this issue are W3C XML Key Management System (Ford et al., 2001), OASIS Security Assertion Mark-up Language (OASIS, 2003a), and IBM and Microsoft WS-Trust (OpenNetwork et al., 2004).

XML Key Management System is a specification that has been subjected to the W3C standardization process and proposes an information format as well as the necessary protocols to convert a PKI (Public-Key Infrastructure) into a

WS so that it will be able to register public/private key pairs; locate public keys; validate keys; revoke keys; and recover keys. This way, the entire PKI is extended to the XML environment, thus allowing delegation of trustworthy decisions to specialized systems. XKMS is presented as the solution for the creation of a trustworthy service that offers all PKI subordinate services but without resolving the inherent issues of the infrastructure: How can a Certification Authority's public key be known with total certainty? Is the CA ascertained identity useful? Known issues with OIDs (Object Identifiers) for automatic processing and their explosive and continuing growth. Since the global public key infrastructure has not a single world-recognized certification authority, it is not clear how to equip the world in order to allow two systems (e.g., WS) that know nothing of one another to establish a trustworthy relationship through a third party on the fly and with no previous off-line communication.

Secure Assertion Mark-up Language is an "OASIS Open Standard" specification developed by OASIS and was delivered in its first version by 2002. This specification is basically divided into two parts: XML schema definition that allows trust assertions (authentication, authorization, or attribute) representation in XML and a client/server protocol to distribute XML authentication, authorization, and attribute assertions.

WS-Trust is another specification deserving mention due to its similarity to XKMS and SAML. WS-Trust defines an XML schema as well as protocols that allow security tokens to be accessed, validated, and exchanged. WS-Trust specifies a framework for trust models that allow WS to securely interoperate by requesting, issuing, and exchanging security tokens.

Privacy

Privacy for the WS users that interacts across multiple domains is a critical issue that the W3C has already defined in its Web Services Architecture Requirements document. In summary, these requirements state that the Web Services Architecture must allow privacy policy statements in P3P (Cranor et al., 2002) to be expressed and advertised, thus enabling WS consumers to access the advertised privacy policy statements. In addition, privacy policies should be able to be delegated and propagated (Ferrari & Thuraisingham, 2004). P3P is a W3C standard that enables WS providers to express their privacy practices in a standard format. These practices are applied to the personal sensitive data that need to be disclosed by the WS consumer when interacting with the WS provider agent.

P3P does not state how the recollected personal data will be conveyed and stored in a secure way. However, it specifies how the WS consumer can locate the privacy practice policies of the WS provider. Enabling the client to express its privacy preferences and the enforcement of the practices specified within a P3P policy are two issues not covered by the P3P specification. Two more standards complement these lacks: APPEL and E-P3P.

APPEL (Cranor, Langheinrich, & Marchiori, 2002) standard complements the P3P specification enabling clients to express their own privacy practice preferences through a set of rules expressing these preferences.

E-P3P "has a well-defined privacy-architecture and semantics that enables an enterprise to internally enforce the E-P3P privacy policies while promising a P3P statement to its customers" (Ashley, Hada, Karjoth, & Schunter, 2002). E-P3P is used by enterprises to describe their internal privacy policies and to enforce the enterprise policy to control access to the collected information.

WS-Privacy (from the IBM and Microsoft initiative and not yet published) is an emerging specification describing a model that allows WS and its requestors to state privacy preferences and organizational privacy practice statements.

Future Trends

In spite of the amount of specifications, standards, and approaches that we have reviewed in this chapter, there are plenty of unresolved security issues that will have to be addressed and standardized in the future:

1. A clear effort should be made from all entities involved in this paradigm in order to unify their criteria and solutions.
2. Services reliability is another aspect that needs further research (Zhang et al., 2004).
3. Web services-based business processes security (Koshutanski & Massacci, 2003).
4. Today, a methodology that accomplishes and considers all possible security issues and defines an organized development process that directs WS deployments in all expected (and unexpected) scenarios does not

exist. This methodology should produce a distributed security framework accomplishing all the security topics related to WS security.

5. End-to-end and large-scale security policy management. Although several major ongoing efforts on the security policy subject exist (WS-Policy, WS-SecurityPolicy, XACML, etc.), they are only specifying ways of representing and conveying policies in XML format. A global security policy management framework should propose solutions to issues such as dynamic establishment of security policies, end-to-end agreements of many-to-many interactions, and security policy version control (Chang et al., 2003).
6. An XML protocol and format vocabulary should be defined so that all WS conforming to it would convey and store their audit data uniformly, respectively. This audit protocol may be created as an extension to existing security protocols such as the WS-Trust or the SAML request/response protocols.

Conclusion

In this chapter, we have reviewed the current WS main research areas, security specifications, standards, and initiatives, and we have shown their diversity. The lack of a global standardization initiative is causing overlap in solutions to similar problems that are put forward. Quoting IBM and Microsoft, "We note that other organizations such as the IETF and ebXML are tackling a related set of problems, and we are pleased there are already formal liaisons between the W3C XML Protocol Working group and its counterparts in both ebXML and IETF" (IBM & Microsoft, 2001). Consequently, all the involved parties will have to make efforts in the future with the purpose of integrating their visions and standards, avoiding conflicts among solutions, and creating a common and global WS (security) framework.

References

Adams, C., & Boeyen, S. (2002, November 22). UDDI and WSDL extensions for Web services: A security framework. *Proceedings of the ACM Workshop on XML Security,* Fairfax, Virginia.

Anderson, A., Proctor, S., & Godik, S. (2004). OASIS XACML profile for Web-services. Retrieved April 30, 2005, from *http://xml.coverpages.org/WSPL-draft-xacmlV04-1.pdf*

Ashley, P., Hada, S., Karjoth, G., & Schunter, M. (2002). E-P3P privacy policies and privacy authorization. *Proceedings of the WPES'02*, Washington, DC, November 21.

Bartel, M., Boyer, J., Fox, B., LaMacchia, B., & Simon, E. (2002, February 12). W3C XML signature syntax and processing. Retrieved April 30, 2005, from *http://www.w3.org/TR/xmldsig-core/*

Bertino, E., Castano, S., & Ferrari, E. (2001). On specifying security policies for Web documents with an XML-based language. *Proceedings of the SACMAT'01,* Chantilly, VA, May 3-4.

Bhatti, R., Bertino, E., Ghafoor, A., & Joshi, J.B.D. (2004). XML-based specification for Web services document security. *IEEE Computer, 37*(4), 41-49.

Blaze, M. (1999). RFC2704 - The keynote trust management system (version 2). Retrieved April 30, 2005, from *http://www.crypto.com/trustmgt/kn.html*

Blaze, M., Feigenbaum, J., & Lacy, J. (1996). Decentralized trust management. *Proceedings IEEE Conference Security and Privacy,* Oakland, CA.

Casati, F., Shan, E., Dayal, U., & Shan, M.-C. (2003). Business-oriented management of Web services. *Communications of the ACM, 46*(10), 25-28.

Chang, S., Chen, Q., & Hsu, M. (2003). Managing security policy in large distributed Web services environment. *Proceedings of the 27th Annual International Computer Software and Applications Conference (COMPSAC'03),* Dallas, TX, November 3-6.

Cranor, L., Langheinrich, M., & Marchiori, M. (2002). A P3P preference exchange language 1.0 (APPEL1.0). Retrieved April 30, 2005, from *http://www.w3.org/TR/P3P-preferences/*

Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M., & Reagle, J. (2002). The platform for privacy preferences 1.0 (P3P1.0) specification. Retrieved April 30, 2005, from *http://www.w3.org/TR/P3P/*

Endrei, M., Ang, J., Arsanjani, A., Chua, S., Comte, P., Krogdahl, P., Luo, M., & Newling, T. (2004). *Patterns: Services oriented architectures and Web services.* IBM Redbook Series. Retrieved from *http://www.redbooks.ibm.com/abstracts/SG246303*

Erl, T. (2004). *Service oriented architecture. A field guide to integrating XML and Web services* (1st ed.). NJ: Prentice Hall PTR.

Ferrari, E., & Thuraisingham, B. (2004). Security and privacy for Web databases and services. *Proceedings Extending Database Technology (EDBT'04),* Heraklion, Crete, Greece, LNCS 2992 (pp. 17-28). Berlin, Heidelberg: Springer-Verlag.

Firesmith, D. (2004). Specifying reusable security requirements. *Journal of Object Technology, 3*(1), 61-75.

Ford, W., Hallam-Baker, P., Fox, B., Dillaway, B., LaMacchia, B., Epstein, J., & Lapp, J. (2001, March 30). W3C XML key management specification (XKMS). Retrieved April 30, 2005, from *http://www.w3.org/TR/xkms/*

Grandison, T., Sloman, M., & College, I. (2000). A survey of trust in Internet applications. *IEEE Comunications Surveys and Tutorials, 3*(4), 2-16.

Iannella, R. (2001, June). Digital rights management (DRM) architectures. *D-Lib Magazine, 7*.

Iannella, R. (2002, September 19). Open digital rights management. Retrieved April 30, 2005, from *http://www.w3.org/TR/odrl/*

IBM & Microsoft. (2001, April 11-12). Web services framework. Retrieved from *http://www.w3.org/2001/03/WSWS-popa/paper51*

IBM & Microsoft. (2002). Security in a Web services world: A proposed architecture and roadmap. Retrieved April 30, 2005, from *http://www-106.ibm.com/developerworks/webservices/library/ws-secmap/*

IDC. (2004). Cautious Web services software adoption continues: IDC expects spending to reach $11 billion by 2008. Retrieved April 30, 2005, from *http://www.idc.com/getdoc.jsp?containerId=pr2004_04_13_090643*

Imamura, T., Dillaway, B., & Simon, E. (2002, December 10). W3C XML encryption syntax and processing. Retrieved April 30, 2005, from *http://www.w3.org/TR/xmlenc-core/*

Koshutanski, H., & Massacci, F. (2003). An access control framework for business processes for Web services. *Proceedings of the ACM Workshop on XML Security,* Fairfax, VA (pp. 15-24).

Li, N., Mitchell, J.C., & Winsborough, W.H. (2002). Design of a role-based trust-management framework. *Proceedings of the 2002 IEEE Symposium on Security and Privacy,* Oakland, CA.

Liberty Alliance Project. (2003). Liberty ID-FF architecture: Overview. v1.2. Retrieved April 30, 2005, from *http://www.projectliberty.org/specs/liberty-idff-arch-overview-v1.2.pdf*

OASIS. (2003a, September 2). Assertions and protocol for the OASIS security assertion markup language (SAML) V1.1. Retrieved April 30, 2005, from *http://www.oasis-open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf*

OASIS. (2003b). eXtensible access control markup language (XACML) version 1.0. Retrieved April 30, 2005, from *http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml*

OASIS. (2004, April 6). Web services security (WS-Security) specification. Retrieved April 30, 2005, from *http://www-106.ibm.com/developerworks/webservices/library/ws-secure*

O'Neill, M., Hallam-Baker, P., Cann, S.M., Shema, M., Simon, E., Watters, P.A., & White, A. (2003). *Web services security.* Berkeley, CA: McGraw-Hill.

OpenNetwork, Layer7, Netegrity, Microsoft, Reactivity, VeriSign, IBM, Oblix, RSASecurity, PingIdentity, Westbridge, & ComputerAssociates. (2004). Web services trust language (WS-Trust). Retrieved April 30, 2005, from *ftp://www6.software.ibm.com/software/developer/library/ws-trust.pdf*

Rescorla, E. (2001). *SSL and TLS. Designing and building secure systems.* Boston: Addison-Wesley.

Sedukhin, I. (2003). End-to-end security for Web services and services oriented architectures. Retrieved from *http:whitepapers.interop.com/detail/RES/1066934613_132.html*

Shandu, R.S., Coyne, E., Feinstein, H., & Youman, C. (1996). Role-based access control models. *IEEE Computer, 29*(2), 38-47.

Skogsrud, H., Benatallah, B., & Casati, F. (2003). Model-driven trust negotiation for Web services. *IEEE Internet Computing,* 45-51.

VeriSign, Microsoft, SonicSoftware, IBM, BEA, & SAP. (2004). Web services policy framework (WS-Policy). Retrieved April 30, 2005, from *ftp://www6.software.ibm.com/software/developer/library/ws-policy.pdf*

W3C. (2004, February 11). Web services architecture. Retrieved April 30, 2005, from *http://www.w3.org/TR/ws-arch/*

WS-I. (2004). Basic security profile version 1.0. Retrieved April 30, 2005, from *http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0.html*

Zhang, L.-J., Zhang, J., & Chung, J.-Y. (2004). An approach to help select trustworthy Web services. *Proceedings of the E-Commerce Technology for Dynamic Business, IEEE International Conference on CEC-East'04,* Beijing, China, September 13-15.

Zimmerman, P. (1994). *PGP user's guide.* Cambridge, MA: MIT Press.

Chapter III

Policies for Web Security Services

Konstantina Stoupa, Aristotle University of Thessaloniki, Greece

Athena Vakali, Aristotle University of Thessaloniki, Greece

Abstract

This chapter analyzes the various types of policies implemented by the Web security services. According to X.800 definition, there are five basic Web security services categories: authentication, non-repudiation, access control, data integrity, and data confidentiality. In this chapter, we discuss access control and data privacy services. Access control services may adopt various models according to the needs of the protected environment. In order to guide the design of access control models, several policy-expressing languages have been standardized. Our contribution is to describe and compare the various models and languages. Data privacy policies are categorized according to their purpose, that is, whether they express promises and preferences, manage the dissemination of privacy preferences, or handle the fulfillment of the privacy promises.

The chapter is enriched with a discussion on the future trends in access control and data privacy.

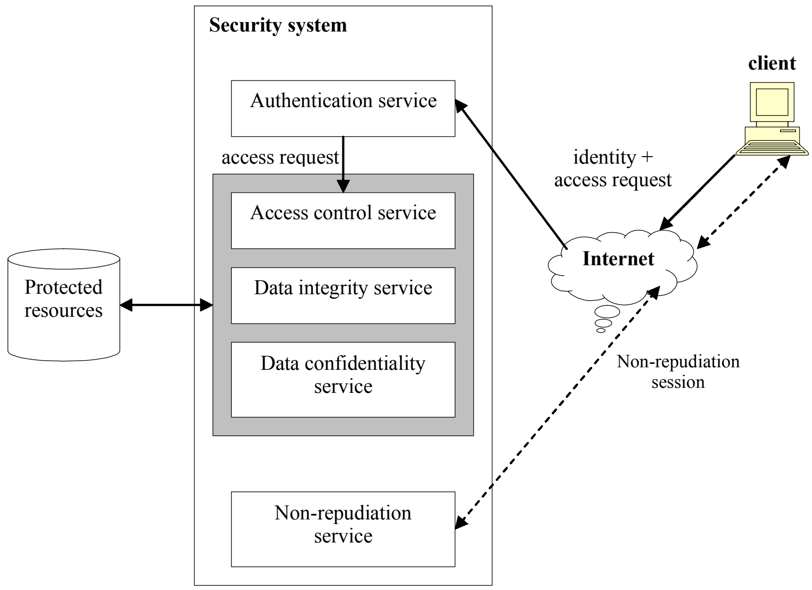
Introduction

Today, users adopt the Internet to complete several business and commercial transactions. The introduction of Web services has enriched this trend. According to Cerami (2002), "a Web service is any service that is available over the Internet, uses a standardized XML messaging system, and is not tied to any one operating system or programming language." As a consequence to their wide adoption, Web services have become the core target of malicious attacks (aiming at either stealing information or causing services and system malfunctions). Therefore, Web-accessed environments need to employ security services to protect their resources (either information or services). Such services enhance the security of data processing, information transferring, and organizational data circulation. Security and protection of Web databases and services have become core research issues, and recent research efforts have focused on these topics (Ferrari & Thuraisingham, 2004; Ferrari, 2004; Thuraisingham, 2002). Overall, security services ensure both secure communication and storage of data, and the proper and continuous execution of Web services.

According to X.800 definition, five basic security services categories exist: *authentication, access control, data confidentiality, data integrity,* and *non-repudiation.* Each of these security services employs security policies which are implemented by security mechanisms (e.g., RFC 2828 glossary). More specifically, we categorize services into:

- **Services for clients' and resources' identities:** verifying the identity of the requesting client and preventing client attempts to deny having accessed a protected resource. Thus, this category involves :
  - **Authentication services:** to verify an identity claimed by (or for) an entity.
  - **Non-repudiation services:** to prevent either sender or receiver from denying a transmitted message.

Figure 1. Security services functionality



- **Services for clients' authorizations over resources:** managing the relationships between clients and protected resources by denying unauthorized actions. This category includes:
  - **Access control services:** for protecting system resources against unauthorized access. They also regulate use of system resources according to a policy, and access is permitted only to authorized entities (users, programs, processes, or other systems) according to that policy.
  - **Data privacy:** for protecting data against unauthorized disclosure (*data confidentiality*) and changes (*data integrity*) including both intentional change or destruction and accidental change or loss by ensuring that changes to data are detectable.

Consider a client (which may be a user, an organization, a procedure, etc.) wants to gain access to protected information stored in the database of a security-based organization. Moreover, some information is personal, and the organization has promised to keep it private according to some privacy policies. Figure 1 depicts these services functionality in such a framework:

1. The initial service triggered is the authentication which verifies the identity of the client. In case it is proven that the client is indeed the one (s)he claims to be, his/her access request is forwarded to the access control service.
2. The access control service consults data integrity and data confidentiality services so as to verify if the client is authorized to gain access and what changes (s)he is allowed to make according to privacy rules governing the storage of protected resources.
3. After the session between the client and the system is terminated, non-repudiation service may be triggered in case the client denies (s)he has gained access to the protected resources, or the organization denies having disclosed the protected data (dashed-line route).

In this chapter, we mainly focus on access control and privacy policy. First, a brief background of our work is given while the following sections refer to access control and data privacy services. Access control is addressed by discussing and comparing policies expressed according to basic access control models. A later section analyzes the core features of data privacy and some well-known frameworks such as P3P, APPEL, EPAL, and E-P3P. The chapter concludes with a discussion of the future trends in security services.

Background

This chapter overviews and presents the most popular current security policies implementing the four security services discussed in the Introduction. Authentication is the first security service that a client interacts with every time (s)he requests a Web service. The identity of a client is proved through a signed identity certificate which is signed by a trusted third party. The most well-known identity certificates are the X.509 certificates associating a subjects' identity with its public key.

Non-repudiation was initially addressed through public key cryptography, and its goal was to produce unforgeable, thus, uncontradictable digital signatures giving the advantage to a receiver of a message to convince a "judge" that the signer sent the message in case of dispute. Such an approach satisfies only the receiver and not the sender. Therefore, more fair schemes have been proposed in some of which the two entities exchange evidences directly and in others through a third party.

Several researchers have worked on the implementation of access control models over time. The earlier models such as those described by Castano, Fugini, Martella, and Samarati (1994) and Denning (1983) support only primitive functionality because they either grant or deny access requests. Those systems were based on cryptography techniques which, as mentioned earlier, cannot provide selective, flexible, and dynamically changing authorizations. Later works (Jajodia, Samarati, Subrahmanian, & Bertino, 1997) introduce mechanisms which support multiple access control policies in the context of non-distributed environments. The next step was the implementation of distributed authorization systems like those discussed by Blaze, Feigenbaum, and Lacy (1996) and Nicomette and Deswarte (1997) where the access control mechanism and the protected data may be stored at different physical locations. Research efforts span several fields such as digital libraries (Adam, Atluri, Bertino, & Ferrari, 2002) and e-business environments (Miklos, 2002).

Data privacy extends access control policies since it defines how the resource will be protected after someone has gained access to it. It is usual in e-commerce applications that personal data given to an enterprise during a session may be forwarded to other cooperating organizations for marketing, advertising, or statistical reasons. Thus, clients become reluctant in using the Internet for completing transactions. Cingil, Dogac, and Azgin (2000) propose a privacy-enriched approach for *personalization*, that is, producing clients' profiles. He created a module for defining RDF-based user profiles according to their click-history which are protected by privacy constraints.

Since there is a great differentiation in the type of protected resources (ranging from data to services), there is a great need for a unified expression format. XML has been considered the most appropriate language for expressing the resources' metadata. Therefore, most of the existing implementations of security policies use XML for expressing the policies, rules, and resources characteristics. Ontologies have also gained ground against simply expressed resources, and, in this context, RDF has also been used in expressing security policies.

Access Control Policies

The Internet and networks, in general, are currently the core media for data and knowledge exchange. A primary issue is to assure authorized access to protected resources located in Web-accessed databases. This task is accomplished through access control services. The history of access control models is not long enough, but it has already presented important improvements, since several models have been introduced to address the current needs of access control.

The basic issues concerning initial access control attempts for expressing policies (consisting of authorizations) are the *subject* (client requesting access), the *object* (protected resource), and the *action* (right or type of access). Therefore, the initial format of authorizations was (subject, object, action) defining which subject can conduct what type of action over what object. However, with the advent of databases, networking, and distributed computing, we have witnessed a phenomenal increase in the automation of organizational tasks covering several physical locations, as well as the computerization of information-related services. Therefore, new ideas have been added into modern access control models like time, tasks, origin, and so forth.

Access Control Models

Over time, several models have tried to address access control. The great difference in policy expression between conventional and modern access control models is the identification of subject. As we shall see later, the subject can be characterized by its identity, its roles in an organization, its tasks, and so on.

Conventional Access Control Models

- **Mandatory Access Control—MAC** (Sandhu & Mason, 1993): it is concerned with enforcing one directional information flow in a lattice of security labels. A MAC policy is expressed in terms of security labels associated with subjects and objects. Each protected resource or each of its elements is associated with a classification label which expresses its sensitivity (i.e., how easily it can be granted access to it). Moreover, each user is equipped with a clearance label which expresses his/her degree of trust. These labels, once assigned, cannot be changed except by the security manager. MAC models provide robust protection mechanisms (based on strict mathematical base) for classified data. However, its lack of flexibility renders MAC model inappropriate for Web-based applications.

- **Discretionary Access Control—DAC:** the main idea of DAC is that the owner of an object, who is usually its creator, has discretionary authority over who else can access that object. Thus, it is about an owner-based administration of access rights appropriate for loose environments. DAC models have been widely used for Web-based applications due to their flexibility. Of course, they do not provide high security assistance as they allow copying of data from one protected object to another which can result in giving access to a copy of data to users who do not have access to the original data. Bertino, Bettini, Ferrari, and Samarati (1996) enrich the DAC authorizations with temporal intervals of validity. After the time expires, authorizations are revoked.

Modern Access Control Models

- **Role-Based Access Control—RBAC** (Sandhu, Coyne, & Feinstein, 1996): Users are assigned one or more roles according to their duties in the organization, and the same holds between roles and permissions. There is also a role hierarchy, where if X>Y then role X inherits the permissions assigned to role Y. Furthermore, administrative roles and permissions are used to manage (create, modify, or delete) the regular roles and permissions, respectively. Many researchers have tried to adopt this model in building access control frameworks and modules (Hitchens & Varahgarajan, 2001; He & Wong, 2000; Kudo & Hada, 2000; Van den Akker, Snell, & Clement, 2001).

  An extension of this model is *Temporal RBAC* (Bertino, Bonatti, & Ferrari, 2001) which supports periodic activations and de-activations of roles, and temporal dependencies among these activations and deactivations.

- **Originator Control—ORCON** (Park & Sandhu, 2002): it is an access control model lying between MAC and DAC. It requires recipients to gain originator's approval for redissemination of an originally disseminated digital object or a new digital object that includes originally distributed digital objects. ORCON is similar to MAC in that access restrictions on original objects are propagated to derived objects. However, ORCON is different from MAC in that policies are modifiable on a subject/object basis, while according to MAC, policies are uniform across all subjects and objects. ORCON is different from DAC in that control privileges on an object can be modifiable only by the originator of the object, while in DAC the owner (recipient) of a derived object can often also change control privileges on the object or on copies of the object. Moreover,

ORCON requires identification of abstract entities like organizations and not users. ORCON can be adopted in open system environments where central security manager is not available.

- **Task-Based Access Control—TBAC** (Thomas & Sandhu, 1993): this model was implemented to address the need for expressing authorizations referring to groups of related activities at several departments of an organization. In this model, every task (a logical unit of work that may consist of several individual subtasks) is assigned an authorization. This model is considered to be appropriate for distributed environments where multiple principles may be involved in the execution of a specific task (thus it can support separation of duty).
- **Owner-Retained Access Control—ORAC** (McCollum, Messing, & Notargiacomo, 1990): it is used for strictly enforced control of access and specific access modes at an individual user level. ORAC provides a stricter, label-based alternative to DAC for user communities where the original owners of data need to retain tight control of the data, as they are disclosed to subjects that may later write the data into other objects.

An Access Control Model for Unknown Users

Since e-commerce has gained ground, we are obliged to consider access control procedures that:

(a) do not offend the client's privacy and anonymity and
(b) are able to work even if the client is unknown.

The answer to those needs is the *digital credentials* which are used in credentials-based access control (Winslett, Ching, Jones, & Slepchin, 1997). Digital credentials have the same use as the physical ones. They are issued by a trusted third party (*credentials authority*) and contain the features of the owner, which are called *attributes*. The management of those credentials is assigned to the client and not the enterprise. Every time a client wants to gain access to a protected resource, it sends its signed credentials. Of course, the client can decide which credentials to disclose, thus, privacy is addressed. Therefore, by adopting this model, we can grant access to subjects with special attributes such as users older than 18 and not with a certain identity.

Table 1 summarizes the basic characteristics of the policies implemented by the above discussed models. It is obvious that subject identification involves the greater differentiations from users to abstract entities, roles, and groups having specific values in their credentials and tasks. Moreover, policies are expressed and handled by specific entities. In MAC, CBAC, and TBAC, a security manager is responsible for expressing policies, while in RBAC and TRBAC, only administrative roles can modify the authorizations related to regular roles. In DAC, ORCON, and ORAC, the owner of the object is responsible for its protection with the difference that, in DAC, someone gaining access to a resource is also considered as its owner. On the other hand, in ORCON and ORAC, only the original owner of the object is responsible for expressing policies or modifying existing ones. Policies are also categorized according to whether they can express dissemination rules (defining access control restrictions after a protected resource is distributed to an authorized subject). Finally, policies can be analyzed against their ability to express strict rules and to trigger authorizations according to temporal constraints.

Table 1. Comparison of policies expressed according to most well-known access control models

	Type of subject	**Policy manager**	**Dissemination control**	**Strict policies**	**Temporal activation of policies**
MAC	User	Security manager	No	Yes	No
DAC	User	current owner	No	No	No
RBAC	Role	Administrative roles	-	Yes	No
TRBAC	Role	Administrative role	-	Yes	Yes
ORCON	Abstract entities or user	Original owner	Yes	Yes	No
TBAC	Tasks	Security manager	No	Yes	No
ORAC	Abstract entities or user	Original owner	Yes	Yes	No
CBAC	Groups according to credentials	Security manager	-	Yes	No

Policies in Standard Access Control Languages

The core of every access control mechanism is a language used for expressing all of the various components of access control policies such as subjects, objects, constraints, and so forth. To date, several languages have been proposed which can support any kind of protected resources in a large variety of environments. Languages in this category include:

- **eXtensible Access Control Markup Language (XACML):** it is an OASIS standard that defines both a policy language and an access control decision request/response language (both written in XML). The policy language is used to describe general access control requirements and has standard extension points for defining new functions, data types, combining logic, and so on. The request/response language permits forming a query in order to ask whether a given action should be allowed (or not) and interprets the result which defines the response.
- **eXtensible rights Markup Language (XrML):** it is a language for digital rights management. By using XrML, the owner of a digital resource can specify who can use the resource (principal) and under which circumstances (constraints). The owner thus produces licenses containing such information that are then given to the appropriate principal. Since XrML is mainly used for protecting digital media, it contains schemas defining the metadata of such resources, such as audio or video files. Of course, it can be easily extended for controlling access to any other type of resource.
- **Open Digital Rights Language (ODRL):** it is a language used for Digital Rights Management (DRM) in open environments. It can express rights for protecting any digital content like electronic publications, digital images, audio and movies, learning objects, computer software, and so forth.
- **WS-Authorization and WS-Policy** (Box et al., 2003): they are specifications of IBM and Microsoft cooperation. WS-Authorization is an unpublished standard that manages the handling of the information used for access control policies and authorizations. WS-policy provides an extensible grammar for expressing the capabilities, requirements, and general characteristics of entities in an XML Web Services-based system.

WS-Policy defines a framework and a model for the expression of these properties as policies.

Web services are provided to requesters under some conditions which are defined by the provider of the Web service. A requester might use this information to decide whether or not to use the service. Sometimes the provider of the service allows the requester to further qualify an expressed policy. The goal of WS-Policy is to provide the mechanisms needed to enable Web services applications to specify policy information. Specifically, this specification defines the following:

- XML-based policy expressions which contain domain specific Web service policy information. Subsequent specifications will define how policy assertions are associated with Web service elements as well as how domain specific policy assertions are expressed.
- A core set of grammar elements to indicate how the contained policy assertions apply.

Table 2 gives a brief comparison of XACML, ODRL, and XrML according to the defined policies features since WS-Authorization is not yet published. Since XACML and ODRL can protect any type of resource, XrML can only support digital media, services, and pieces of information. XACML and XrML provide fine-grained protection since they can even protect attributes of elements. As it concerns the access control model, only XACML is role-based while the

Table 2. Expressing policies using the most well-known access control languages

	XACML	**XrML**	**ODRL**
Type of object	Any type of resource	Digital media, services, pieces of information	Any physical or digital content
Protection granularity	Fine-grained	Fine-grained	Coarse-grained
Type of subject	User, group, role (through subject-category)	User with his private key	User identity
Permission/denials	Both	Permissions	Permissions
Access modes	Any action	Predefined actions with extension possibility	Any action

other two adopt user identity to identify subjects. XACML seems to be the most complete of all since policies can contain both permissions and denials. Moreover, XrML seems to be quite inflexible since it supports only predefined access modes.

Data Privacy Policies

The Web has caused an information revolution since it transfers large amounts of information. The ease of information access and high availability of personal data made it more tempting to interested entities (such as business, governments, or individuals). Those entities try to intrude on personal information and thus offend individuals' privacy. The inherently open and non-deterministic nature of the Web seems to exacerbate the privacy problem. Its open character allows every unknown user to access information. Moreover, the transferred information is leakage-prone since the receiver may disclose it to other parties without the consent of the sender.

Some steps have been taken in order to enforce clients' privacy, which are discussed in Rezgui, Bouguettaya, and Eltoweissy (2003). The most flexible and widely accepted is the enforcement of privacy policies which has the form of self regulations expressed by e-businesses. Those policies may be certified (through trust seals) or not.

Policies can be used for expressing privacy promises, privacy preferences, privacy enforcement rules, or dissemination rules. In this section, we will discuss the most well-known frameworks for expressing such policies.

Policies for Privacy Promises Expression

The first standard ever implemented for expressing privacy promises was the Privacy Preferences Project (P3P) which has been designed by W3C. It is about a platform enabling Web sites to express their privacy practices in a standard format which can be forwarded to users. Since P3P is a policy expression framework, it does not contain mechanisms for transferring data or for securing data in transit or storage. Thus, it should be used in conjunction with other technologies like HTTP, cryptography, and so forth.

For the P3P to work, user agents are needed which are placed into client's browsers and are responsible for informing users on site practices and automating decision making based on these practices. The privacy practices can have both machine-readable format (for decision-making processes) and human-readable formats (for users accessing the site).

The P3P specification is XML-based; thus, policies and collected data are expressed in XML. Unfortunately, P3P has some limitations since it supports a standard set of purposes, recipients, and data categories. Thus, some companies may not be able to define explicitly their privacy practices.

Policies for Privacy Preferences Expression

So far, we have discussed the way we can express and enforce privacy policies, but we did not analyze how a customer interacting with sites can express his/her privacy preferences so as the user agent makes automated or semi-automated decisions regarding the acceptability of machine-readable privacy policies. The most well-known language is the *P3P Preference Exchange Language (APPEL)* which complements P3P specification. Using this language, a user can express his/her preferences in a set of preference rules (called a rule set). Thus, this language will be used to allow users to import preference rule sets created by other parties and to transport their own rule sets files between multiple user agents. User agent implementors might also choose to use this language (or some easily-derived variation) to encode user preferences for use by the rule evaluators that serve as the decision-making components of their user agents. APPEL is used by Lategan and Olivier (2002) to form a Chinese wall security policy.

Although APPEL seems like quite an attractive language for expressing languages, it has two basic problems: (a) it does not allow the specification of rules defining what is acceptable. It only offers specify-unacceptable rules, and (b) even the specify-unacceptable rules are constrained in terms of the logical expressions that could be provided in a single rule. These two problems are solved by *XPref* (Agrawal, Kiernan, Srikant, & Xu, 2003), which is an XPath-based language for expressing preferences extending APPEL. The rule behavior part of the APPEL rules remains untouched, but the rule body consists of XPath expressions which have the ability to combine many rules defining both acceptable and unacceptable policies.

Policies for Privacy Promises Fulfillment

P3P was the first platform defined for addressing the privacy problem. Its main characteristic was to make privacy statements which could be sent to the client. It did not contain any mechanism for enforcing those policies, that is, guarantee that they are satisfied. Lately, a language has been designed in order to address that goal: *the Platform for Enterprise Privacy Practices (E-P3P)*. E-P3P uses rules to internally manage personal information. Those rules define which cooperating entity can have access to collected personal information. The rules are defined according to the client's preferences, for example, if a client has bought an item from an e-store, and he/she also has consent to the dissemination of his/her personal information to statistical services, then the e-store can authorize a cooperating statistical organization to gain access to the information.

E-P3P is a fine-grained privacy model which is used to enforce privacy statements expressed in P3P (Ashley, Hada, & Karjoth, 2002). Most building blocks of E-P3P policies have been introduced as features of access control policies. Policies define what data users can perform what actions for what purposes on which data categories. On the other hand, when an entity requests access to protected personal information, it should send a request which will be checked against existing rules.

E-P3P policies contain a number of rules which allow or deny actions on *data-categories* by *user-categories* for certain *purposes* under certain *conditions* while mandating certain *obligations*. Each rule of a policy is associated with an integer depicting its precedence. Moreover, purposes and data categories are organized into hierarchies, reducing the number of rules in an E-P3P policy.

Every time a subject wants access to protected resources, it should send a request to the E-P3P enabled system containing the following fields: (a) data-category, (b) purpose, (c) data user, and (d) action. This request is checked against existing rules, and the subject gets an answer informing whether his/her request is allowed or denied and according to which rule (rule-id) and a list of his/her obligations in case access is allowed. Requests may be both simple and complex.

Provisions and obligations are used by several frameworks to support data privacy. Of course, some mechanisms should also be implemented for monitoring them and taking compensatory actions in case they are ignored. Bettini, Jajodia, Wang, and Wijesekera (2002) introduce such a tool where every client gaining access is associated with a level of trustworthiness. In case he/she fails to fulfill his privacy promises, one of the following actions may take place: (a) decreasing his/her trustworthiness, (b) replacing unfulfilled obligations with more costly alternatives, (c) taking punitive actions, or (d) terminating the policy in-force. Moreover, obligation chains can be formed, implying that the fulfillment of an obligation requires the fulfillment of another.

Policies for Disseminating Private Information

The most well-known language for expressing policies of this category is *Enterprise Privacy Authorization Language (EPAL)*. EPAL is an XML-based, IBM-proposed formal language for writing and enforcing enterprise privacy policies (Backes, Bagga, Karjoth, & Schunter, 2004). It is based on E-P3P, therefore, on hierarchies, since all of the involved entities form tree structures. Although EPAL is based on E-P3P, it does not support the *precedence* element in the definition of rules. In EPAL, rules belonging to the same policy are sorted by descending precedence.

EPAL is an interoperability language for exchanging privacy policies in a structured format between applications or enterprises. Therefore, it can work as an extension of APPEL since it cannot express client preferences. Furthermore, EPAL is not based on specific data schemas; thus, it can express policies involving any data type satisfying the needs of every enterprise. Moreover, it can be used by legacy systems due to the use of data abstractions. The produced policies are both computer- and human-readable like the P3P policies.

The interesting feature of this platform is the exchange of privacy policies between enterprises. Before any interaction, cooperating organizations exchange a vocabulary defining the elements participating into privacy policies. Afterward, one organization may request access to data collected by another one.

Mont, Pearson, and Bramhal (2003) ensure privacy through XML-expressed *sticky policies*, that is, policies combined with the associated private information into a digital package. This method is appropriate in cases of multi-party interactions where one organization may disclose one person's personal information to another. The whole project is based on *Tracing Authorities (TA)* which can be implemented by the user or another trusted entity, and they are responsible for the distribution of information. Thus, the user sends an encrypted digital package to the requesting organization which, in order to decrypt it, should send to the TA its identity and its promise to fulfill the policies. TA informs the user, asks for his/her consent, and afterward, it sends the key to the requesting organization. Of course, multiple TAs could be engaged for more security. In such a case, each TA keeps a subkey, and in order to decrypt the credentials, the organization should contact all of the involved TAs to collect all of the needed subkeys.

All these transactions are audited by the TA. So, in case of an illegal redistribution of private information, the guilty could be found. The only limitation of this technique is that it cannot prevent the illegal redistribution of information, but it can only detect it and find the guilty party.

Future Trends and Conclusions

The future trends in Web services security take into consideration open environments which can be accessed by unknown users. This can be achieved through the use of digital credentials which associate clients with some of their features that influence the access control procedure. Moreover, great concern is given to client's privacy, and current attempts focus on preserving the privacy of personal information after it has been disclosed by the owner.

Modern research efforts concerning access control involve protection of Internet-accessed environments where users are unknown. In such cases, their characteristics, roles, or credentials should be contained in a certificate issued and signed by an external certification authority. Therefore, a standard governing the syntax of such certificates seems to be mandatory. One trend is to extend identity certificates such as X.509 certificates in order to also include access control information. A framework protecting an Internet-accessed environment using certificates is proposed by Stoupa and Vakali (2003). Their certificates are produced by an external authority, and they are translated into an understandable format by the protecting mechanism to process their contents. Moreover, Johnston, Mudumbai, and Thompson (1998) introduce authorizations certificates containing use conditions which can be used in environments where users are known but are highly dispersed. Similar environments try to protect the distributed authorization mechanism discussed by Dai and Alves-Foss (2001).

Efforts toward data privacy basically concern control and management of privacy rules. Great attention is given on keeping the rules effective. Companies that collect private information should be controlled and punished if they do not keep their privacy promises. Therefore, third parties are implemented which should be informed about organizations' promises and keep track of actions that may offend the privacy of clients' personal information.

In this chapter, we have tried to introduce the security needs of Web-enabled environments and how they can be addressed by security policies. Those policies have been categorized into authentication, non-repudiation, access control, and data privacy policies. According to this categorization, several models have been analyzed, and some well-known implementations have been discussed and compared. Special interest has been given to credentials-based access control services which can manage unknown users.

References

Adam, N.R., Atluri, V., Bertino, E., & Ferrari, E. (2002, March-April). A content-based authorization model for digital libraries. *IEEE Transactions on Knowledge and Data Engineering, 14*(2), 296-315.

Agrawal, R., Kiernan, J., Srikant, R., & Xu, Y. (2003). An Xpath-based preference language for P3P. *Proceedings of the 12th International World-Wide Web Conference,* Budapest, Hungary.

Ashley, P., Hada, S., & Karjoth, G. (2002). E-P3P privacy policies and privacy authorization. *Proceedings of the ACM Workshop on Privacy in the Electronic Society,* Washington, DC.

Backes, M., Bagga, W., Karjoth, G., & Schunter, M. (2004). Efficient comparison of enterprise privacy policies. *Proceedings of the ACM Symposium on Applied Computing*, Nicosia, Cyprus.

Bertino, E., Bettini, C., Ferrari, E., & Samarati, P. (1996). A temporal access control mechanism for database systems. *IEEE Transactions on Knowledge and Data Engineering, 8*(1), 67-81.

Bertino, E., Bonatti, P.A., & Ferrari, E. (2001). TRBAC: A temporal role-based access control model. *ACM Transactions on Information and System Security, 4*(3), 191-233.

Bertino, E., Castano, S., & Ferrari, E. (2001, May-June). Securing XML documents with Author-X. *IEEE Internet Computing, 5*(3), 21-31.

Bettini, C., Jajodia, S., Wang, X.S., & Wijesekera, D. (2002). Obligation monitoring in policy management. *Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Network,* Monterey, California.

Blaze, M., Feigenbaum, J., & Lacy, J. (1996). Decentralized trust management. *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, Oakland, CA.

Box, D., Curbera, F., Hondo, M., Kaler, C., Langworthy, D., Nadalin, A., Nagaratnam, N., Nottingham, M., Von Riegen, C., & Shewchuk, J. (2003). Web services policy framework (WS-Policy). Retrieved April 30, 2005, from *http://xml.coverpages.org/ws-policyV11.pdf*

Castano, S., Fugini, M., Martella, G., & Samarati, P. (1994). *Database security.* Reading, MA: Addison-Wesley.

Cerami, E. (2002). *Web services essentials.* O'Reilly and Associates.

Cingil, I., Dogac, A., & Azgin, A. (2000, August). A broader approach to personalization. *Communications of the ACM, 43*(8), 136-141.

Dai, J., & Alves-Foss, J. (2001). Certificate based authorization simulation system. *Proceedings of the 25th Annual International Computer Software and Applications Conference,* Chicago.

Damiani, E., De Capitani di Vimercati, S., Paraboschi, S., & Samarati, P. (2000). Design and implementation of an access control processor for XML documents. *Proceedings of the 9th World Wide Web Conference*, Amsterdam, The Netherlands.

Denning, D.E. (1983). *Cryptography and data security.* Reading, MA: Addison-Wesley.

Ferraiolo, D.F., Sandhu, R.S., Gavrilla, S.I., Kuhn, D.R., & Chandramouli, R. (2001). Proposed NIST standard for role-based access control. *ACM Transactions on Information and System Security, 4*(3), 224-274.

eXtensible Access Control Markup Language (XACML) Committee Specification, Version 1.1. (n.d). Retrieved April 30, 2005, from *http://www.oasis-open.org/committees/xacml/repository/cs-xacml-specification-1.1.pdf*

eXtensible rights Markup Language (XrML) 2.0 Specification (n.d.). Retrieved April 30, 2005, from *http://www.xrml.org*

Ferrari, E. (2004). Protection of Web documents: Principles and research issues. *Proceedings of the 1st International Workshop on Clustering Information over the Web*, Heraklion, Greece.

Ferrari, E., Adam, N.R., Atluri, V., Bertino, E., & Capuozzo, U. (2002). An authorization system for digital libraries. *The International Journal on Very Large Databases, 11*(1), 58-67.

Ferrari, E., & Thuraisingham, B. (2004). Security and privacy for Web databases and services. *Proceedings of the 9th International Conference on Extending Database Technology*, Heraklion, Greece.

He, H., & Wong, R.K. (2000). A role-based access control model for XML repositories. *Proceedings of the 1st IEEE International Conference on Web Information Systems Engineering,* Hong Kong, China.

Hitchens, M., & Varahgarajan, V. (2001). RBAC for XML document stores. *Lecture Notes of Computer Science, 2229*, 131-143.

Jajodia, S., Samarati, P., Subrahmanian, V.S., & Bertino, E. (1997). A unified framework for enforcing multiple access control policies. *SIGMOD Record (ACM Special Interest Group on Management of Data), 26*(2), 474-485.

Johnston, W., Mudumbai, S., & Thompson, M. (1998). Authorization and attribute certificates for widely distributed access control. *Proceedings of the 7th IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises,* Stanford, CA.

Kudo, M., & Hada, S. (2000). XML document security based on provisional authorization. *Proceedings of the 7th ACM Conference on Computer and Communications Security*, Athens, Greece.

Lategan, F.A., & Olivier, M.S. (2002). A Chinese wall approach to privacy policies for the Web. *Proceedings of the 26th Annual International Computer Software and Applications Conference,* Oxford, UK.

McCollum, C.J., Messing, J.R., & Notargiacomo, L. (1990). Beyond the pale of MAC and DAC: Defining new forms of access control. *Proceedings of the 1990 IEEE Symposium on Security and Privacy,* Oakland, CA.

Miklos, Z. (2002). A decentralized authorization mechanism for e-business applications. *Proceedings of the IEEE International Workshop on Trust and Privacy in Digital Business,* Aix-en-Provence, France.

Mont, M.C., Pearson, S., & Bramhal, P. (2003). Towards accountable management of identity and privacy: Sticky policies and enforceable tracing services. *Proceedings of the 14th International Workshop on Database and Expert Systems Applications,* Prague, Czech Republic.

Nicomette, V., & Deswarte, Y. (1997). An authorization scheme for distributed object systems. *Proceedings of the IEEE Symposium on Security and Privacy,* Oakland, CA.

Open Digital Rights Language (ODRL) Specification, Version 1.1. Retrieved April 30, 2005, from *http://odrl.net/1.1/ODRL-11.pdf*

P3P Preference Exchange Language 1.0 (APPEL1.0). Retrieved April 30, 2005, from *http://www.w3.org/TR/2002/WD-P3P-preferences-20020415*

Park, J.S., & Sandhu, R. (1999). Smart certificates: Extending X.509 for secure attribute services on the Web. *Proceedings of the 22nd National Information Systems Security Conference*, Crystal City, VA.

Park, J., & Sandhu, R. (2002). Originator control in usage control. *Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Networks,* Monterey, CA.

The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. Retrieved April 30, 2005, from *http://www.w3.org/TR/2002/REC-P3P-20020416/*

Qin, L., & Atluri, V. (2003, Month 00-00). *Concept-level access control for the Semantic Web.* ACM Workshop on XML Security, Fairfax, VA.

Rezgui, A., Bouguettaya, A., & Eltoweissy, M.Y. (2003, November-December). Privacy on the Web: Facts, challenges, and solutions. *IEEE Security and Privacy,* 40-49.

Sandhu, R.S., Coyne, E.J., & Feinstein, H.L. (1996, February). Role-based access control models. *IEEE Computer,* 38-47.

Sandhu, R.S., & Mason, G. (1993, November). Lattice-based access control models. *IEEE Computer,* 9-19

Stoupa, K., & Vakali, A. (2003). An XML-based language for access control specifications in an RBAC environment. *Proceedings of the 2003 IEEE Conference on Systems, Man & Cybernetics,* Washington, DC.

Thomas, R.K., & Sandhu, R.S. (1993). Task-based authorization: A paradigm for flexible and adaptable access control in distributed applications. *Proceedings of the 16th NIST-NCSC National Computer Security Conference,* Baltimore.

Thuraisingham, B. (2000). Data and application security: Development and directions. *Proceedings of the 26th Annual International Computer Software and Applications Conference,* Schoorl, The Netherlands.

Thuraisingham, B. (2002). Building secure survivable Semantic Web. *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence,* Washington, DC.

Van den Akker, T., Snell, Q.O., & Clement, M.J. (2001). The YGuard access control model: Set-based access control. *Proceedings of the Sixth ACM Symposium on Access Control Models and Technologies*, Chantilly, VA.

Winslett, M., Ching, N., Jones, V., & Slepchin, I. (1997). Using digital credentials on the World-Wide Web. *Journal on Computer Security, 5*, 255-267.

Witchert, M., Ingham, D., & Caughey, S. (1999). Non-repudiation evidence generation for CORBA using XML. *Proceedings of the 15th Annual Computer Security Applications Conference*, Phoenix, AZ.

Chapter IV

Data Confidentiality on the Semantic Web: Is There an Inference Problem?

Csilla Farkas, University of South Carolina, USA

Abstract

This chapter investigates the threat of unwanted Semantic Web inferences. We survey the current efforts to detect and remove unwanted inferences, identify research gaps, and recommend future research directions. We begin with a brief overview of Semantic Web technologies and reasoning methods, followed by a description of the inference problem in traditional databases. In the context of the Semantic Web, we study two types of inferences: (1) entailments defined by the formal semantics of the Resource Description Framework (RDF) and the RDF Schema (RDFS) and (2) inferences supported by semantic languages like the Web Ontology Language (OWL). We compare the Semantic Web inferences to the inferences studied in traditional databases. We show that the inference problem exists on the Semantic Web and that existing security methods do not fully prevent indirect data disclosure via inference channels.

Introduction

The emergence of standardized languages, such as the eXtensible Markup Language (W3C, 2004a), the Resource Description Framework (W3C, 2004b), and the Web Ontology Language (W3C, 2004c), supports automated data management. These languages provide simple syntax and precise semantics that are understandable to both humans and machines. The envisioned Semantic Web (Berners-Lee, Hendler, & Lassila, 2001; Hendler, Berners-Lee, & Miller, 2002) and the applications taking advantage of the Semantic Web will be built upon these languages. A necessary requirement for these future applications is to provide information security and privacy.

Existing security solutions for the Web target specific areas like trust management, secure Web services, access control models for XML, and Web privacy (see Thuraisingham, 2002 for an overview). A promising new research trend aims to incorporate semantics in security models like semantic-aware access control and policy specification. Although the number of research and development efforts to provide security for the Semantic Web is increasing, only a few researchers consider the inference problem in this context (Farkas & Jajodia, 2002).

Inferences over semantically enhanced data and metadata play a fundamental role on the Semantic Web. Indirect disclosures resulting from the inference capabilities of the Semantic Web are similar to the inference problem studied in statistical and relational databases (Farkas & Jajodia, 2002; Jajodia & Meadows, 1995). However, the characteristics of these two environments differ from the perspectives of (1) data completeness, (2) scope of data control, (3) data models, (4) amount of data (scalability), and (5) data quality. These characteristics affect not only the detection of indirect data accesses but also the applicable removal methods. For example, in traditional databases, removal of an inference channel is usually performed by limiting accesses to data used to derive unwanted inference. However, in the open and decentralized environment of the Semantic Web, some of the data yielding unwanted inferences may be outside of the protected domain. In this case, removal of the inference channel may not be possible by limiting data accesses. New approaches like leakage of misleading information need to be considered.

The goal of this chapter is to evaluate the risk of unwanted inferences in the context of the Semantic Web. Our claim is that the risk of such inferences has increased due to large-scale, semantically enhanced, and automated data processing (Stoica & Farkas, 2002, 2004; Frakas & Stoica, 2003). We compare the inference threat on the Semantic Web to the inference problem studied in traditional databases. We study two types of inferences: (1) entailments defined by the formal semantics of the Resource Description Framework (RDF) and the RDF Schema (RDFS) and (2) inferences supported by semantic languages like the Web Ontology Language (OWL). Indirect data disclosure may occur due to the existence of replicated data with inconsistent security classification and inferences that disclose disallowed data or data associations. Existing access control models that are applicable to Semantic Web data and metadata do not prevent such indirect disclosures, thus are unable to protect against inference-based attacks. Since inferences are considered a fundamental activity on the Semantic Web, we believe that it is necessary to consider their impact on security.

The organization of the chapter is as follows. The Semantic Web section contains a brief overview of the Semantic Web technologies. The Database Inference Problem section presents the inference problem in traditional databases. The Inferences Problem on the Semantic Web section discusses possible inference threats on the Semantic Web, including RDF-based inferences and ontology-driven inferences. The Security Analysis and Future Trends section lists the distinguishing characteristics of the Semantic Web inference problem, outlines prevention methods, and identifies future research areas. The last section, a summary, concludes the chapter.

The Semantic Web

This section gives a brief overview of the Semantic Web and Web inference engines. For detailed description of the related concepts, the reader should consult the Web pages of the World Wide Web Consortium (http://www.w3c.org) and the Semantic Web Community Portal (http://www.semanticweb.org).

XML, RDF, and Ontology Languages

The eXtensible Markup Language (XML), XML schema, Resource Description Framework (RDF), and RDF schema are the basic components of the

Semantic Web. XML (W3C, 2004a) separates data content from its presentation. XML syntax supports interoperation between heterogeneous domains. Recent research considers XML from the perspective of data management (Abiteboul, Buneman, & Suciu, 2000; Buneman et al., 2003; Thuraisingham, 2002). These works aim to develop concepts and tools for XML that are similar to the ones in traditional database systems. XML provides simple syntax without formal semantics. RDF and RDFS (W3C, 2004b) are intended to describe metadata about Web resources. They represent a knowledge-oriented approach with formal semantics and the notion of entailment (RDF Semantics, 2004). Ontology languages (Erdman & Studer, 1999; Horrocks, Patel-Schneider, & van Harmelen, 2003; McGuinness, Fikes, Stein, & Hendler, 2002) layered on top of XML and RDF enable more expressive (compared to RDF and RDFS) and provable deductions. Ontologies provide a common ground among heterogeneous domains. The Web Ontology Language (OWL) is the latest standard of the World Wide Web Consortium (W3C, 2004c).

Semantic Web Inference

The main distinguishing characteristics of the Semantic Web, in contrast to the current Web, are the presence of machine understandable representations of data semantics and inferences derived from them. Researchers like Shah (1998), Sheth (1999), and Patel and Sheth (2001) address the need of modeling data semantics. The main application targeted by these research activities is intelligent data integration.

RDF, RDFS, and ontology languages support formal semantics and provable deductions. The problem of building formal models of Web inferences has been targeted by several researchers (Boley, 2000; Grosof, Horrocks, Volz, & Decker, 2003; Horrocks, 2002; Horrocks et al., 2003; McGuinness et al., 2002; RuleML, 2004). In addition to the theoretical models, software tools like Swish (Klyne, 2003), Haskell (2004), Racer (2004), and SweetJess (Grosof, Horrocks, Volz, & Decker, 2003) have been implemented.

Privacy Issues in the Context of the Semantic Web

At the end of the 1990s, due to the increased data availability and the use of the World Wide Web, concerns about personal data security surfaced (Biskup

& Bruggemann, 1989; Meeks, 1997). Intelligent data integration, supported by the Semantic Web, will further increase the privacy risk. In addition to direct disclosure of personal data, indirect disclosures via inference channels also have to be considered. Automated technologies like software agent platforms may also generate privacy violations (Farkas & Huhns, 2002).

Research that considers indirect privacy violations via data mining (Clifton, Kantarcioglu, & Vaidya, 2002; Oliveira, Zaiane, & Saygin, 2004) and data linkage (Subramaniam & Yang, 2003; Wang et al., 2004) has emerged recently. Privacy preserving data mining is motivated by the need to protect sensitive data as data mining applications grow. Privacy concerns include the personal information about individual users and information about the activities of a group of users. Work on data linkage addresses the problem of disclosing personal data from aggregate information or from separately-released, non-confidential data of an individual. Prevention methods are based on techniques like perturbation and generalization.

Database Inference Problem

Inference channels in databases occur when sensitive information can be disclosed from non-sensitive data and metadata. The inference problem involves the detection and removal of inference channels. Jajodia and Meadows (1995) and Farkas and Jajodia (2002) present overviews of the inference problem in relational databases. Metadata in databases may correspond to database constraints, such as database dependencies and integrity constraints, or outside information, such as domain knowledge and query correlations. Depending on the level of accuracy by which the sensitive information is revealed, we can distinguish between full or partial disclosure.

To illustrate a simple inference channel via integrity constraints, consider a Credit Card database over schema *(Name, Available credit, Max. credit, Balance)*, where the *Balance* information (i.e., how much a customer owes) is confidential. Clearly, by revealing the *Available credit* and *Max. credit* data and the integrity constraint *Balance = Max credit – Available credit*, any user could calculate the *Balance.*

Inference Control Strategies

While no general solution to the inference problem exists (Jajodia & Meadows, 1995), several methods have been developed that provide protection against specific types of unwanted inferences. Inference channel detection and removal methods can be characterized into two main groups based on the time of the inference channel analysis: (1) during database-design time and (2) during query-processing time.

The main advantages of the first approach is that it is computationally less expensive than the second approach since it only considers the database schema and the corresponding constraints without the actual instances. In addition, the analysis does not slow the performance of the database usage. However, this approach may result in overclassification because an inference channel may not materialize in a particular database instance. Inference channel detection during query-processing provides maximal data availability because all disclosed data can be evaluated to verify the existence of an inference channel. However, there are two main disadvantages in this approach. First, it is usually more expensive and time consuming than the design time approach and affects system usage. Second, the refusal and answer of the queries themselves may create an inference channel.

Inference Problem Modeling

Several approaches have been developed by the database security community to model the inference problem. In particular, two of these approaches seem to be relevant to the Semantic Web inference problem: (1) logic-based and (2) semantic relationship modeling-based.

Logic-based approaches, like the ones presented by Thuraisingham (1987) and Brodsky, Farkas, and Jajodia (2000), have been shown to be promising for detecting inference channels and establishing formal properties of the inference algorithms. They model the database as a collection of facts and apply metadata (inference rules) to these facts to deduce additional information. This approach is related to the Semantic Web knowledge representation using Description and First Order Logic (Boley, 2000, 2001; Grosof, Gandhe, & Finin, 2003; Grosof, Horrocks, Volz, & Decker, 2003; Horrocks, 2002; Sowa, 2000). The advantages of this approach are the clear formalism and the decidability results, allowing the establishment of assurance of the techniques.

However, logic-based systems have high complexity, making the approach expensive for large applications like database systems or the Web.

The second approach (Hinke, Delugach, & Wolf, 1997) uses semantic modeling techniques to represent relationships among data items and detects inference channels. Inference detection is performed on semantic or conceptual graphs that allow reasoning about database entities and domain knowledge. Inference channels are detected by identifying multiple paths between entities such that some of the paths are not visible to the users. This approach shows similarities to the semantic relationship modeling of Web data (Shah & Sheth, 1998; Sheth, 1999).

The Inferences Problem on the Semantic Web

We consider inferences defined by the formal semantics of the languages used for building the Semantic Web. We start out with RDF-based inferences, followed by ontology driven inferences.

RDF-Based Inferences

RDF statements are binary relational assertions. These assertions may entail other assertions that may not be explicitly represented in the original set of RDF statements. The model-theoretic semantics of RDF defines the set of possible worlds based on a set of RDF assertions. Intuitively, a set *S* of RDF statements entails an RDF statement *s*, if for any world, whenever *S* is true then *s* must also be true. Entailment generates data that may not be explicitly stored but could be inferred from RDF data. From the security perspective, this derivable information must also be incorporated in the authorization framework. Two fundamental questions must be answered: (1) how to automatically classify the derived data and (2) how to detect unauthorized data accesses via derived data.

RDF Semantics (2004) present entailment rules for RDF and RDFS. The authors distinguish among simple, RDF, RDFS, and extensional entailment rules. These rules, along with the RDF statements, allow formal inferences. Researchers, like Sowa (2000) and Boley (2000, 2001), developed logic-based approaches to model RDF, RDFS, and possible inferences. For example, Sowa (2000) showed that the expressive power of RDF is the same as existential-conjunctive subset of first-order logic.

Existing inference engines operate on the *sub-class-of* and the *sub-property-of* constructs of RDFS. These simple inferences may lead to security violations as mentioned above. For example, assume that the following three assertions of our model (for simplicity, we omitted the namespaces):

1. *Companies-With-Deficit* is an RDFS *class*
2. *Airlines* is a sub class of *Companies-With-Deficit*
3. *United* is an instance of *Airlines*

These three assertions together entail that United is a company that has a deficit, that is, United rdf:type Companies-With-Deficit. Assuming that United wanted to keep its financial status confidential, the newly generated assertion violates the security requirements. However, without considering the entailment, the three assertions alone do not violate the security.

RDF and Metadata

In addition to the entailment rules defined by RDF and RDFS semantics, modeling complex relationships among the database components may also lead to unwanted inferences. Consider the scenario when the RDF data contains the following facts: (1) Ann is the parent of Mary; (2) Mary is the parent of Pete; (3) Ann is female; and (4) Ann is the grandmother of Pete. Assume that Pete would like to keep it confidential that Ann is his grandmother. A simple solution is to classify the fourth fact "Ann is the grandmother of Pete" confidential. Unfortunately, while this controls the direct disclosure of the fact that Pete is Ann's grandchild, it does not prevent the inference engine to infer this fact given the other three assertions and the following two rules:

1. **if** X is the parent of Y **and** Y is the parent of Z **then** X is the grandparent of Z
2. **if** X is a grandparent of Y **and** X is female **then** X is the grandmother of Y

To our best knowledge, no authorization model exists that would prevent unwanted inferences via RDF entailment.

Ontology-Based Inferences

Initial studies of the inference problem via ontologies on the Semantic Web are given in Farkas and Stoica (2003) and Stoica and Farkas (2004). The authors target unwanted inferences in semantically enhanced and publicly available XML documents. The XML documents provide simple syntax. Mappings from the XML documents to ontology concept hierarchies provide an interpretation for the syntactic XML constructs. The authors present methods to generalize XML tags according to the ISA relationship in the concept hierarchy. The basic assumption is that if two tags can be generalized to the same concept, then they are semantically related. The authors set a number of measurements, such as number of generalization steps, relative difference between the tags, and the specificity of the concept, in order to measure the correctness of the assumption that the two tags are related. The inference detection is performed at the schema level. Based on the users' preferences, probable inference channels can be further evaluated by data-level analysis.

The authors study two specific problems. The first problem is the detection of replicated data in distributed XML repositories where data may have different syntax and structure. Security violations occur when the replicated data have inconsistent or contradictory security classifications. The second problem concerns the detection of confidential data associations that can be deduced from publicly available associations. Both modules are incorporated in Ontology Guided XML Security Engine (Oxsegin).

Replicated Data Inference

In Stoica and Farkas (2004), the authors propose a method to detect replicated XML data with conflicting security classifications. Their main technical contribution is the development of the Probabilistic Inference Engine used by Oxsegin. The inference engine operates on DTD files and uses the ISA relationship in the corresponding ontology to identify XML elements that can be generalized to the same concept. Users can specify the number of permitted generalization steps, eliminating the possibility that all tags are unified under a

Figure 1. Inconsistent security classification (from Stoica & Farkas, 2004)

XML File 1 DOD repository **Figure 1.a**	**XML File 2 John's computer** **Figure 1.b**
<?xml version="1.0"?>	<?xml version="1.0"?>
<cryptoTools> ………………………….. **S**	<academic> ………………………….. **C**
<document> ………………………….. **S**	<paper> …………………………… **C**
<title>CA1059</title>	<title>CA1059</name>
<author>John Smith</author>	<writer>John Smith</writer>
<project>P987HY5</project>	</paper>
</document>	</academic>
</cryptoTools>	

common concept. If two elements can be generalized to the same concept, they are called ontologically equivalent. Security violations occur when two ontologically equivalent tags have inconsistent or conflicting security classifications. These tags are marked with a security violation pointer (SVP). A confidence level coefficient (CLC) is attached to every SVP, representing the level of assurance of the derived security breach.

Consider the following example from the paper of Stoica and Farkas (2004). Figure 1 shows two XML documents over the same data but with different syntax and structure. The first XML document is classified *Secret*. The second XML document is classified *Confidential*. A human observer of these two XML documents would conclude that they represent the same data, that is, discover the relationships between document and paper, author and writer, and that the security classifications of the replicas are inconsistent. The authors present a probabilistic inference engine and security module that reach the same conclusions for a predefined set of XML documents, corresponding ontologies, and security policy.

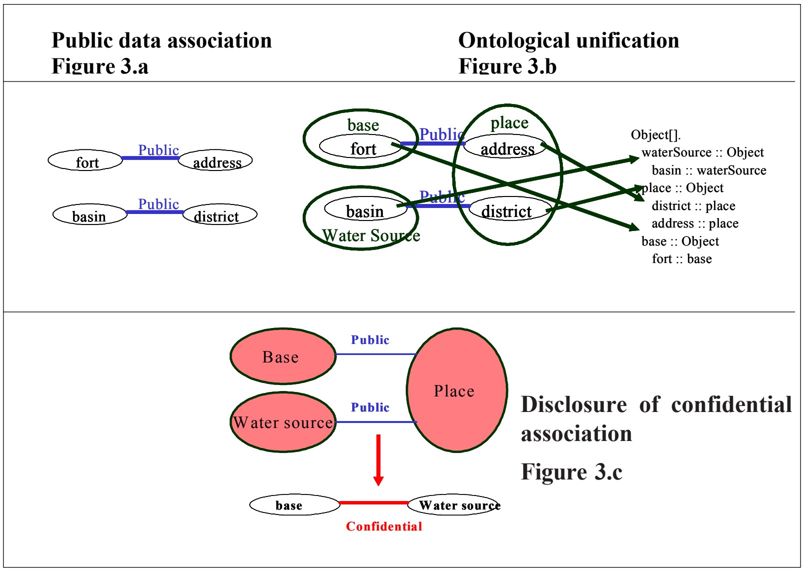
Correlated Data Inference

In Farkas and Stoica (2003), the authors propose a method to detect unauthorized disclosure of sensitive associations. They propose a Correlated Inference Procedure to detect data associations such that the security classifications of the associations are inconsistent. The procedure incorporates syntactic and structural differences of the XML documents and uses the ISA

Figure 2. Disclosure of secret association (from Farkas & Stoica, 2003)

Air-show information **Figure 2.a**	**Drinking water basins** **Figure 2.b**	**Critical Infrastructure** **Figure 2.c**
<?xml version="1.0"?> <show> ………………….. **P** <fort> Base_X </fort> <address> District_Y </address> </show>	<?xml version="1.0"?> <waterMap> …… **P** <district>District_Y </district> <basin>Basin_Z </basin> </waterMap>	<?xml version="1.0"?> <infrastructure> ………….. **S** <base> Base_X </base> <waterSource> Basin_Z </waterSource> </infrastructure>

Figure 3. Illustration of discovering confidential association



relationship to generalize tags to common concepts. Similar to the inference engine for replicated data, the method works on DTD files.

To represent this problem, consider the three XML files shown in Figure 2. The air-show information (Figure 2a) and the drinking water basin (Figure 2b) documents are classified public. The critical infrastructure file (Figure 2c) contains the sensitive association between an army base and its water reservoir. This association is classified *secret*.

Use of the ontology shown in Figure 3b allows the generation of a public-level association between *base* and *water source* thus revealing the secret information. Figure 3 illustrates the unification procedure.

Security Analysis and Future Trends

Current studies of inference control address only the ISA relationship among the concepts. Further research is needed to integrate complex relationships and their properties. In addition, security solutions need to be tailored to the special characteristics of the Semantic Web inferences.

Characteristics of the Semantic Web Inferences

Semantic Web inferences have characteristics similar to the inference problem studied in traditional databases. However, there are also several differences. These differences are best viewed from the perspectives of (1) data completeness, (2) scope of data control, (3) data model, (4) amount of data (scalability), and (5) data quality.

Traditional inference control methods protect data explicitly stored in a database and classified according to a security policy. In contrast, the information content of the Semantic Web is dynamic and incomplete, thus limiting the applicability of the existing methods. Further, the inference algorithm properties, with respect to soundness and completeness, need to be re-evaluated in the context of the Web context. Also, since the inference procedure may generate new data that is not explicitly stored in the database, mechanisms that automatically assign security classifications to such data must be developed.

The openness of the Web leads to another distinguishing characteristic, the scope of the controlled area. In traditional databases, it is generally assumed that the security officer has control over all data items leading to an undesired inference. This assumption does not hold true for Web data where inferences may use data residing in someone else's domain. Traditional prevention methods, based on limiting access to data leading to unwanted inferences, are not applicable in this case.

Another distinguishing characteristic of Semantic Web inferences is that they reason over heterogeneous data like XML, RDF, relational, and so forth. This will further limit the applicability of traditional prevention methods that focused mainly on relational databases.

Scalability is a critical aspect for any practical application. One of the most overwhelming features of the Web is its size. Assuming the existence of ontologies and semantically annotated data, inference engines must process large amount of documents.

Additional characteristics include the problem of unambiguously identifying protection objects, for example, XML substructures, RDF resources, and the quality and trustworthiness of the data and metadata used for the inference. These issues must be carefully evaluated when establishing the assurance of the inference control. Current research addresses only a small set of these problems.

Future Trends

In the previous section, we presented some of the currently investigated aspects of Semantic Web inferences. Tools to support large-scale metadata extractions and semantic annotations of heterogeneous data are being developed. Some of the problems mentioned in this chapter are currently being analyzed from the perspectives of data integration and Semantic Web inferencing. Collaborations among these researchers and security experts are necessary to achieve high-assurance, practical systems.

We need to develop security models and methods that will not restrain interoperation but satisfy security and privacy requirements. The first line of defense of data security is access control. However, to answer the question of what information must be protected and from whom, we need to consider the identity of the protection object and that it may not be explicitly stored but can be derived from the existing data. Authorization models that identify protection objects based on their semantics rather than syntax and are capable of handling extensional and intentional data within a single framework need to be developed.

Another crucial aspect of the Semantic Web inference problem is how to remove detected inference channels. If controlled data items are part of the unwanted inferences, then the removal of an inference channel is possible by restricting access to these data items. However, if the inference channel is raised by non-controlled public data, then removal of the inference channel becomes difficult and costly. Our main assumption is that we only have control over the data and metadata contained within our domain.

Studying Web inferences requires that we answer several crucial questions. Some of these questions are currently being studied from the perspective of providing efficient inference capabilities. However, they have not been studied from the perspective of security and privacy. In addition to the previous discussion, the following questions must be answered.

- *What is the assurance of the developed model and how do we deal with inconsistencies?* Unbounded Web-based applications do not conform to the traditional database security models for security assurance and require new definitions for concepts like completeness and correctness. Furthermore, trustworthiness of sites and distributed data depositories have a major affect on evaluating correctness and completeness.
- *How do we deal with semantic inconsistencies and ambiguities?* Data and metadata available on the Web may be incorrect, leading to inconsistencies and contradictions. Simple conflict resolution methods, like denial takes precedence, are not sufficient to resolve conflict. Methods that allow the users to judge the correctness of the data need to be incorporated in the inference engine to support flexible conflict resolution.
- *Do we need to control the disclosure of data semantics?* Data semantics play a significant role in providing interoperations. However, it is realistic to assume that levels of interoperations may differ among the participants, requiring that they can access semantic information at different levels. For example, a company may allow others to access general information about the company, but only trusted organizations are allowed to access specific organizational metadata.
- *How can we prevent unwanted inferences?* In general, there is no silver bullet to remove unwanted inferences. However, it has been shown that solutions exist for specific problems. Unwanted inference prevention on the Semantic Web may involve techniques ranging from information hiding to the introduction of noise or may have to be addressed in a non-information technology manner like increased physical security.

Conclusion

This chapter presented a brief overview of current Semantic Web technologies and related security models. The main focus of the chapter is the evaluation of the security threats from unwanted inferences. Our aim is to encourage researchers to develop security paradigms that do not unnecessarily limit interoperability while satisfy security and privacy requirements. Our belief is that the inference problem is indeed a security threat against data confidentiality and privacy on the Semantic Web. However, no security model exists that fully addresses this threat. Future research to develop security models for Semantic Web data and metadata, and to detect and remove unwanted inferences is needed. The works of Stoica and Farkas (2002, 2003, 2004) present an initial analysis of the related problems; however, the authors only address inferences via the ISA relationship. Security methods, capable of handling complex, possibly interdomain relationships (Sheth, 1999; Patel & Sheth, 2001), need to be developed. Further, formal assurance of these methods needs to be established with respect to soundness and completeness. Development of a satisfactory solution for the Semantic Web inference problem requires the collaboration between security experts, Semantic Web developers, and domain experts.

Acknowledgments

This work was partially supported by the National Science Foundation under grant number IIS-0237782. We owe very special thanks to Dr. Bhavani Thuraisingham for her encouragement and comments on the draft of this manuscript.

References

Abiteboul, S., Buneman, P., & Suciu, D. (2000). *Data on the Web: From relations to semistructured data and XML.* San Francisco: Morgan Kaufmann.

Berners-Lee, T., Hendler, J., & Lassila, O. (2001, May) The Semantic Web. *Scientific American, 284*(5), 34-43.

Biskup, J., & Bruggemann, H.H. (1989). The personal model of data: Towards a privacy oriented information system. *Proceedings of the Fifth International Conference of Data Engineering* (pp. 348-355).

Boley, H. (2000). Relationship between logic programming and XML. *Proceedings of the 14th Workshop Logische Programmierung*.

Boley, H. (2001). *The rule markup language: RDF-XML data model, XML schema hierarchy, and XSL transformations.* Invited Talk, INAP2001. Tokyo.

Brodsky, A., Farkas, C., & Jajodia, S. (2000). Secure databases: Constraints, inference channels and monitoring disclosures. *IEEE Transactions on Knowledge and Data Engineering, 12*(6), 900-919.

Buneman, P., Davidson, S., Fan, W., Hara, C., & Tar, W.C. (2003). Reasoning about keys for XML. *Information Systems, 28*(8), 1037-1063.

Clifton, C., Kantarcioglu, M., & Vaidya, J. (2002). Defining privacy for data mining. *Proceedings of the National Science Foundation Workshop on Next Generation Data Mining* (pp. 126-133).

Erdman, M., & Studer, R. (1999). Ontologies as conceptual model for XML documents. *Proceedings of the 12th Workshop for Knowledge, Acquisition, Modeling and Management*, Banff, Canada, October.

Farkas, C., & Huhns, M. (2002). Making agents secure on the Semantic Web. *IEEE Internet Computing,* 76-79.

Farkas, C., & Jajodia, S. (2002). The inference problem: A survey. *SIGKDD Explorations, Special Issue on Privacy and Security, 4*(2), 6-12.

Farkas, C., & Stoica, A. (2003). Correlated data inference in ontology guided XML security engine. *Proceedings of the IFIP 17th WG 11.3 Working Conference on Data and Application Security*.

Grosof, B.N., Gandhe, M.D., & Finin, T. (2003). SweetJess: Inferencing in situated courteous RuleML via translation to and from Jess Rules. Retrieved May 1, 2005, from *http://ebusiness.mit.edu/bgrosof/paps/sweetjess-wp-050203.pdf*

Grosof, B.N., Horrocks, I., Volz, R., & Decker, S. (2003). Description logic programs: Combining logic programs with description logic. *Proceedings of the 12th International World Wide Web Conference.*

Haskell A Purely Functional Language. (2004). Retrieved May 1, 2005, from *http://www.haskell.org*

Hendler, J., Berners-Lee, T., & Miller, E. (2002). Integrating applications on the Semantic Web. *Journal of the Institute of Electrical Engineers of Japan, 122*(10), 676-680.

Hinke, T.H., Delugach, H.S., & Wolf, R.P. (1997). Protecting databases from inference attacks. *Computers and Security, 16*(8), 687-708.

Horrocks, I. (2002). Reasoning with expressive description logics: Theory and practice. *Proceedings of the 19th International Conference on Automated Deduction*.

Horrocks, I., Patel-Schneider, P.F., & van Harmelen, F. (2003). From SHIQ and RDF to OWL: The making of a Web ontology language. *Journal of Web Semantics, 1*(1), 7-26

Jajodia, S., & Meadows, C. (1995). Inference problems in multilevel secure database management systems. *Information Security: An integrated collection of essays* (pp. 570-584). Los Alamitos, CA: IEEE Computer Society Press.

Klyne, G. (2003). Semantic Web inference using Haskell. Retrieved May 1, 2005, from *http://www.ninebynine.org/Software/swish-0.1.html*

McGuinness, D.L., Fikes, R., Stein, L.A., & Hendler, J. (2001). DAML-ONT: An ontology language for Semantic Web. In D. Fensel, J. Hendler, Lieberman, & W. Wahlster (Eds.), *Spinning the Semantic Web: Bringing the World Wide Web to its full potential*. Cambridge, MA: MIT Press.

Meeks, B.N. (1997). Privacy lost, anytime, anywhere. *Communications of the ACM, 40*(8), 11-13.

Oliveira, S.R.M., Zaiane, O.R., & Saygin, Y. (2004). Secure association rule mining. *Proceedings of the Eighth Pacific-Asian Conference on Knowledge Discovery and Data Mining* (pp. 74-85).

Patel, S., & Sheth, A. (2001). *Planning and optimizing semantic information requests on heterogeneous information sources using semantically modeled domain and resource characteristics* (LSDIS Technical Report). University of Georgia.

Racer: Semantic Middleware for Industrial Projects based on RDF/OWL and W3C Standards (2004). Retrieved May 1, 2005, from *http://www.sts.tu-harburg.de/~r.f.moeller/racer/*

RDF Semantics. (2004). Retrieved May 1, 2005, from *http://www.w3.org/TR/rdf-mt/*

Rule Markup Language Initiative. (2004). RuleML for the Semantic Web. Retrieved May 1, 2005, from *http://www.dfki.uni-kl.de/~boley/RuleML-RDF-OntoWeb/*

Shah, K., & Sheth, A. (1998). Logical information modeling of Web-accessible heterogeneous digital assets. *Proceedings of the Forum on Research and Technology Advances in Digital Libraries (ADL),* Santa Barbara, CA.

Sheth, A. (1999). Changing focus on interoperability: From system, syntax, structure to semantics. In M. Goodchild, M. Egenhofer, R. Fegeas, & C. Kottman (Eds.), *Interoperating geographic information systems* (pp. 5-30). Boston: Kluwer Academic.

Sowa, J.F. (2000). *Knowledge representation: Logical, philosophical and computational foundations.* Pacific Grove, CA: Brooks Cole.

Stoica, A., & Farkas, C. (2002). Secure XML views. *Proceedings of the 16th IFIP WG11.3 Working Conference on Database and Application Security*.

Stoica, A., & Farkas, C. (2004). Ontology guided security engine. *Journal of Intelligent Information Systems*, *23*(3), 2009-223.

Subramaniam, H., & Yang, Z. (2003). *Report on DIMACS working group on privacy/confidentiality of health data* (Technical Report). Rutgers University, DIMACS Center, CoRE Building, Piscataway, NJ.

Thuraisingham, B. (1987). Security checking in relational database management systems augmented with inference engines. *Computers and Security, 6*, 479-492.

Thuraisingham, B. (2002). *XML databases and the Semantic Web.* Boca Raton, FL: CRC Press.

Wang, D.W., Liau, C.J., Hsu, T.S., & Chen, J.K. (2004). On the damage and compensation of privacy leakage. *Proceedings of the 18th Annual IFIP WG 11.3 Working Conference on Data and Applications Security*.

W3C. (2004a). Extensible markup language (XML). Retrieved May 1, 2005, from *http://www.w3.org/XML/*

W3C. (2004b). Resource description framework (RDF). Retrieved May 1, 2005, from *http://www.w3.org/RDF/*

W3C. (2004c). Web ontology language (OWL). Retrieved May 1, 2005, from *http://www.w3.org/2001/sw/WebOnt/*

Chapter V

Secure Semantic Grids

Bhavani Thuraisingham, University of Texas at Dallas, USA

Latifur Khan, University of Texas at Dallas, USA

Abstract

This chapter first describes the developments in semantic grids and then discusses the security aspects. Semantic grids integrate Semantic Web and grid technologies. They also integrate knowledge management with Semantic Web and the grids. There is much interest on applying the semantic grid for many applications in science and engineering. Therefore, it is critical that the semantic grids be secure. We will also discuss Semantic Web mining and privacy implications relevant to semantic grids.

Introduction

The Semantic Web is essentially the next generation Web. Tim Berners-Lee's vision is to have machine understandable Web pages (Berners-Lee & Fischetti, 2000). Subsequently, he invented the Semantic Web. Semantic Web essentially integrates many technologies including Web database management, Web services, and infrastructures. Furthermore, Berners-Lee has designed the layered architecture for the Semantic Web, including layers for XML (eXtensible Markup Language), RDF (Resource Description Framework), and Web Rules and Query. Many developments have been reported on the developments of the Semantic Web. Ontologies and ontology management techniques have been developed. Much progress has been made on information integration.

While progress was made on the Semantic Web, the global grid community started focusing on building the next generation grid. The term grid evolved from power grids where millions of power lines have to be supported by the grid. Similarly with the computing grid, millions of resources that form the grid have to be allocated efficiently to the computationally intensive tasks. Typically, these tasks are those carried out for scientists and engineers.

The semantic grid is a new term that has evolved over the last couple of years. It essentially integrates Semantic Webs with the global grids. That is, grid computing has to be made more intelligent. Integrating ontologies and Semantic Web technologies into grid computing will enable the development of semantic grids. There has been some progress on semantic grids (see for example IEEE, 2004). However, security for semantic grid has received little attention. To achieve security for the semantic grid, we need security for the Semantic Web and secure grids. It is only recently that security has been examined for Semantic Webs and grids (Bertino, 2004; Thuraisingham, 2004a). This chapter attempts to incorporate security for semantic grids.

The organization of this chapter is as follows. We discuss semantic grids including a discussion of Semantic Webs, grids, and integration issues. Knowledge management will also be discussed. Then, we discuss security issues including secure Semantic Web, secure grids, secure integration, and secure knowledge management. Some further considerations in related topics such as Semantic Web mining and privacy considerations will be then be discussed.

Semantic Grids

Overview

Several technologies have to work together to develop the semantic grid. These include the Semantic Web, the global grid, and the integration of the two technologies. The Semantic Web is essentially about machine understandable

Web pages. The global grid supports computationally intensive applications for scientists and engineers and determines how to allocate the various computing resources. Information integration is about integrating the disparate information sources on the Web.

This section discusses the underlying technologies for the Semantic Web. An overview of the Semantic Web, including a discussion of ontologies, will then be given. Next, grid computing will be discussed and integration of the Semantic Web and the grid will follow. The chapter then briefly discuss knowledge management and its relationship to semantic grids.

Semantic Web

Layered Architecture

The availability of data on the Web is constantly expanding. Most of this data is semi-structured, consisting of text (HTML) and multimedia (sound, video, and image). Overall, this data constitutes the largest body of information ever accessible to any individual. The problem with this data is that they are not machine-understandable. It is very hard to automate anything on the Web, and because of the volume of information the Web contains, it is not possible to manage it manually. The solution proposed here is to use metadata to describe the data contained on the Web (Berners-Lee & Fischetti, 2000) (i.e., Semantic Web). The Semantic Web is based on a vision of Tim Berners-Lee.

The Semantic Web is an extension of today's Web technology; it has the ability to make Web resources accessible by their semantic contents rather than merely by keywords and their syntactic forms. Due to its well-established mechanisms for expressing machine-interpretable information, Web services and information previously available for human consumption can be created in a well-defined, structured format from which machines can comprehend, process, and interoperate in an open, distributed computing environment. This proves to be quite advantageous with respect to data collection; intelligent software entities, or agents, can effectively search the Web for items of interest, which they can determine with new semantic knowledge.

Berners-Lee suggests enriching the Web by machine-processable information which supports users in their tasks. For instance, today's search engines are already quite powerful but still return often too large or inadequate lists of hits.

Machine-processable information can point the search engine to the relevant pages and can thus improve both precision and recall. To reach this goal, the Semantic Web will be built up in different levels in an incremental fashion: (a) Unicode/URI, (b) XML/Name Spaces/XML Schema, (c) RDF/RDF schema, (d) Ontology vocabulary, (e) logic, (f) proof, and (g) trust. A common syntax is provided using the first two layers (i.e., Unicode, XML). URIs provide a standard way to refer to entities while XML fixes a notation for describing labeled trees.

- **Resource Description Framework (RDF):** RDF is a critical part of the Semantic Web. The RDF data model is a syntax-neutral way of representing RDF expressions. The basic data model consists of three object types, resources, properties, and statements. An RDF model can be represented by a directed graph. Therefore, any node in the RDF model can have multiple children and multiple parents.

  RDF is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information on the Web. RDF emphasizes facilities which enable the automated processing of Web resources. Since RDF is designed to describe the resources and the relationship among them without assumption, the definition mechanism should be domain neutral and can be applied to any domain. Here are the following key terms in RDF (1999):
  - **Resource:** A Resource is anything that can have a URI (Unified Resource Identifier); this includes the entire Web's pages, as well as individual elements of an RDF document. For instance, when we reach a Web page with a hyperlink "http://weather.yahoo.com/regional/USTX.html", that is a resource. Each resource is uniquely identified by a URI.
  - **Property:** A Property is a Resource that has a name and can be used as a property; for example, a company has many properties such as its name, locations, and so forth.
  - **Statement:** A Statement consists of the combination of a Resource, a Property, and a value.

Let us assume that there is a virtual company with the URI http://www.virtualsite.com; the founder of the company has his own homepage http:/

/www.virtualsite.com/~qchen; and the full name of the founder is Qing, Chen. The main technical support e-mail address is main@virtualsite.com; and the company introduction is composed by the founder. From these statements and relationships among them, we can draw the RDF model (RDF Schema).

RDF model is an edge directed graph. A node in the graph represents a subject or an object, which can be URI, empty node, or literal value while the link from the subject to an object is the predicate, which can be a URI. Any node in the RDF model can have multiple children and multiple parents. For most cases, there are roots in the RDF model, which do not have any parents.

The next layers in the Semantic Web architecture are the ontology vocabulary and logic. Today, Semantic Web considers these two levels as one single level due the support of logical axiom by ontologies. Proof and trust are the remaining layers that facilitate validity check of statements made in the Semantic Web. Since these two layers demand further significant research, in this chapter, we will present more details about only ontology vocabulary layer.

- **Ontologies:** An ontology is an explicit specification of a conceptualization (Gruber, 1993); the term is borrowed from philosophy where ontology is a systematic account of being or existence from the Greek *ontos* (being). In the context of the Semantic Web, what "exists" is that which can be represented. In the context of artificial intelligence, we can describe the ontology of a domain by defining a set of representational terms. In such an ontology, definitions are used to associate the names of entities in the universe of discourse (e.g., classes, relations, functions, or other objects) with human-readable text describing what the names mean and formal axioms that constrain the interpretation and focus the well-formed use of these terms. Formally, an ontology is the statement of a logical theory.

  Ontologies play an important role in providing an ability to model, represent, and exchange formal conceptualization as well as information of particular domains in a precise, machine-understandable form. Standard, predefined domain-specific ontologies can be shared and reused. Interrelationships among the concepts describe a target world. An ontology can be constructed in two ways, domain dependent and generic. Generic ontologies attempt to provide a general framework for all (or most) categories encountered by human existence (Miller, 1995). Generic ontologies are generally very large, difficult to build, and void of ample details. A domain-dependent ontology will contain a conceptual model about some domain (say sports domain) (Khan & McLeod, 2000; Khan, McLeod, & Hovy, 2004; Hovy, 1998). The ontology can be used to represent information or knowledge about sports and various sports concepts and the relationships that are assumed to be held between them. For example, various sports can be characterized as either team or individual sports, or both. Also, there are specific sports, which can be played individually as well as in a team environment.

- **OWL-S:** OWL-S is the specific technology used to define the Web ontology (OWL, 2004). To make use of a Web service, a software agent needs a computer-interpretable description of the service and the means by which it is accessed. An important goal for Semantic Web markup languages, then, is to establish a framework within which these descriptions are made and shared. Web sites should be able to employ a set of basic classes and properties for declaring and describing services, and the ontology structuring mechanisms of OWL provide the appropriate framework within which to do this.

  Moreover, to enable one Web service to automatically locate and utilize services offered by other Web services, a mechanism must be in place to describe the properties of each Web service involved in terms of its properties, functionalities, constraints, and interfaces such as inputs and outputs. OWL-S provides the solution, providing facilities for describing service capabilities, properties, pre-/post-conditions, and input/output specifications.

  OWL-S is ontology for describing Web services, enabling users and software agents to automatically discover, invoke, compose, and monitor Web resources offering services under specified constraints. OWL-S supplies Web service providers with a core set of markup language constructs for describing the properties and capabilities of their Web services in unambiguous, computer-interpretable form. OWL-S markup of Web services will facilitate the automation of Web service tasks, including automated Web service discovery, execution, composition, and interoperation.

Global Grid

The global grid evolved from high performance computing technologies which include parallel processing technologies. Parallel processing is a subject that has been around for a while. The area has developed significantly from single processor systems to multi-processor systems. Multi-processor systems could be distributed systems, or they could be centralized systems with shared memory multi-processors or with shared-nothing multi-processors. There has been a lot of work on using parallel architectures for database processing (see for example IEEE, 1989). While considerable work was carried out, these systems did not take off commercially until recently. This is partly due to the explosion of data warehousing and data mining technologies where performance of query algorithms is critical.

In a parallel database system, the various operations and functions are executed in parallel. While research on parallel database systems began in the 1970s, it is only recently that we are seeing these systems used for commercial applications. This is partly due to the explosion of data warehousing and data mining technologies where performance of query algorithms is critical.

Let us consider a query operation which involves a join operation between two relations. If these relations are to be sorted first before the join, then the sorting can be done in parallel. We can take it a step further and execute a single join operation with multiple processors. Note that multiple tuples are involved in a join operation from both relations. Join operations between the tuples may be executed in parallel.

Many of the commercial database system vendors are now marketing parallel database management technology. This is an area we can expect to grow significantly over the next decade. One of the major challenges here is the scalability of various algorithms for functions such as data warehousing and data mining. Many of the data mining techniques are computationally intensive. Appropriate hardware and software are needed to scale the data mining techniques. Database vendors are using parallel processing machines to carry out data mining.

More recently, grid computing is making a major impact on high performance computing. The idea is to use clusters of computing systems including database systems to carry out various functions in parallel. The challenge is to allocate resources to various tasks so that high performance computing applications can be executed efficiently. Many applications for science and engineering such as bioinformatics and geoinformatics require substantial computing power. Therefore, clusters of computers are assigned to specific jobs rather than using mainframe machines. Cluster computing is being adopted by major vendors including Oracle and IBM. For example, some of the recent versions of Oracle products take advantage of cluster database management. This is the essence of grid computing.

Integration

Semantic grid essentially integrates both the Semantic Web and grid computing. It is essentially about building a layer above the global grid that will manage the resources, the metadata, as well as support machine-understandable Web pages. It also integrates Web services with the global grid. Web services are services such as resource management services, directory services, publishing services, subscription services, and various other services that are provided on the Web.

We briefly discuss a possible concept of operation for the Semantic Web. Suppose a scientist is writing an article and wishes to use results from an online experiment. (S)he can log onto the semantic grid, utilize the Semantic Web to search for various material on the subject. (S)he could then use the global grid to carry out some experiments and then use the results in the paper. Essentially, the scientist is utilizing the grid and Semantic Web technologies not only to write the research paper but also to produce results possibly in real-time. One can also envision a scenario where the semantic information manager essentially takes over and automates the process by invoking the Semantic Web as well as the global grid and then give pertinent results and information to the scientist. Essentially, not only is the user taking advantage of the Semantic Web for the information, (s)he is also using the grid for the computing power needed to carry out the tasks.

Knowledge Management

Knowledge management is a closely related technology for the Semantic Web and the semantic grid. Knowledge management is about taking effective actions on the information that is produced by a corporation. It also includes protecting the intellectual property; sharing information, products, and services; as well as enhancing the effectiveness and competitiveness of an organization. Knowledge management is a cyclic process. First of all, knowledge has to be created and represented. That is, knowledge from humans has to be transformed into some media. Knowledge then is manipulated so that it can be disseminated in a useful format. Knowledge has to be sustained, and actions have to be taken. The cycle continues by creating more knowledge.

Knowledge management components include strategies, processes, metrics, and technologies (Morey et al., 2001). Knowledge management strategies are about setting up plans for knowledge management. Processes are the actual means of sharing and enhancing the knowledge. Metrics are about measuring the knowledge such as the number of patents obtained by a corporation. Finally, technologies are the tools used for knowledge management. These tools could range from database systems to the Semantic Web (Thuraisingham, 2002).

Essentially, Semantic Web can be regarded as a tool for knowledge management. With the Web, knowledge can be shared effectively. On the other hand, knowledge management, such as creating and managing ontologies, is an effective tool for the Semantic Web. Knowledge management can also be used to determine which resources to use for computing and, subsequently, enhance the power of the grid. Similarly, grid can be used to enhance knowledge management by determining which jobs to execute and which clusters to give priority.

Secure Semantic Grid

Overview

If the applications are to be executed securely, the semantic grid has to be secure. This means that the nodes of the grid must have permission to execute various applications. Since applications are executed at different nodes, if there is some malicious code, then the system must ensure that the vulnerabilities are not propagated during application execution. Applications must also possess credentials to utilize various resources. There are many similarities between grid and the Web. Semantic grid is similar to the Semantic Web where knowledge and information management would play a role in allocating resources as well as application execution. Security for the semantic grid is yet to be investigated. Closely related to the grid is what is known as the Cyber infrastructure. Cyber infrastructure is essentially providing the data and computing resources for scientists and engineers to carry out their tasks that are computationally intensive (Atkins, 2002). Little work has been reported on security for the Cyber Infrastructure.

This section discusses security for the semantic grid. In particular, it focuses on secure Semantic Web, secure grid, and the secure integration of various technologies. Next, it discusses secure Semantic Web and is followed by a discussion on secure grid. Secure integration and secure knowledge management follow. Finally, further details on this topic are also given in Thuraisingham (2005).

Secure Semantic Web

Earlier, we discussed the Semantic Web. In this section, we will examine security issues. For the Semantic Web to be secure, we need to ensure that all of the layers of the Semantic Web are secure. This includes secure XML, Secure RDF, Secure Ontologies, and Secure information integration. While work has been carried out on securing XML (Bertino et al., 2002; Bertino et al., 2004), little work has been carried out on securing RDF documents as well as secure information interoperability. Some preliminary directions for RDF security as well as for expressing policies in RDF are presented in Carminati et al. (2004). Some ideas on secure Semantic Web are given in Thuraisingham (2004a).

In the case of XML security, the challenges include the granularity of classification as well as secure publication of XML documents. That is, should one classify the entire XML document, or should one classify portions of the document? Access control policies for various components of the XML documents have been investigated extensively by Bertino et al. (2002). Role-based access control for XML documents has also been investigated. Bertino et al. have also investigated the publication of XML documents with untrusted third-party publishers. They have explored enforcing authenticity, integrity, and completeness. They are also exploring the issues on querying encrypted databases to enforce confidentiality.

In the case of securing RDF documents, the question is: can we apply the techniques developed for XML security? Note that in addition to the XML syntax, RDF also enforces semantics. This makes it more difficult to enforce security as one could infer highly sensitive and private information from the unclassified and public information. Some work on designing security architecture for the RDF model has been explored in Carminati et al. (2004). Furthermore, like XML, RDF can also be used to specify security policies.

Earlier in this chapter, we discussed ontologies as well as OWL-S. The questions are: How can we enforce access control on ontologies? How can security properties be specifies in OWL-S? Furthermore, can we use ontologies to specify security policies and assist in secure information integration? This is a major challenge and needs further investigation.

Finally, we need to ensure that the heterogeneous information sources are integrated securely. That is, can we use the ideas presented in secure federated data management for integrating the various information sources securely? How can the various security policies be integrated? How can we enforce trust management policies across the different sites? For a discussion of secure federated data management and the integration of security policies, we refer to Thuraisingham (1994).

This section has discussed briefly how security could be enforced across the layers of the Semantic Web. We started with XML security and then migrated to RDF security. Finally, we discussed secure ontologies and information integration. There are many challenges, and we need a focused research program on securer Semantic Web.

Secure Global Grid

This chapter has discussed the global grid and high performance computing. We also discussed parallel processing and data management. In this section, we discuss secure high performance computing and the global grid. Some work has been carried out on integrating both security and parallel processing as well as security and real-time processing. In the case of secure parallel data management, the challenge is to enforce the security policies while executing the database operations in parallel. For example, if subqueries Q1 and Q2 are to be executed in parallel, then each subquery will enforce the relevant access control policies. However, there may be policies to be enforced that involve data processed by both queries. The system must ensure that such policies are taken into consideration when executing the queries.

In the case of real-time computing and secure computing, the challenge is to enforce security, yet meet the timing constraints. This is a conflict, as access control techniques may be time consuming. Therefore, we need to come up with flexible policies. In some situations, we need to ensure that the timing constraints are met, and, in other situations, we need to ensure that security is maintained.

Securing the global grid will include secure parallel processing as well as secure real-time computing. In addition, the grid must ensure that the clusters enforce the security policies. Furthermore, policies across the clusters have to be enforced as well. The grid must also ensure that there are no malicious attacks that could result in denial of service or the compromise of classified data. Secure grid computing is in its infancy. There were some interesting discussions on this topic at the ACM Symposium on Access Control Models and Technologies (Bertino, 2004). We need a research program on securing the global grid.

Secure Integration

Just as the semantic grid integrates both Semantic Web and the global grid, the secure Semantic Web should integrate the secure Semantic Web with the secure global grid. Now, both Semantic Web and the global grid are not mature technologies. That is, the two technologies are evolving. Furthermore, it is only recently that security has been examined for the grid and the Semantic Web. Therefore, developing a secure semantic grid is a long-term goal. Nevertheless, if the information has to be shared across organizations securely and if the computing resources are allocated and executed securely, then we need to develop a secure semantic grid.

Secure integration of the Semantic Web and the grid has many challenges. Let us consider the concept of operation of the semantic grid discussed earlier. Suppose a scientist is writing an article and wishes to give access control only to certain individuals. Furthermore, (s)he wishes to use results from an online experiment. (S)he can log onto the semantic grid, utilize the Semantic Web to search for various material on the subject depending on the access controls that (s)he has, use the global grid to carry out some experiments again depending on the execute permissions that (s)he has, and complete the article using the online results and grant access to the various individuals who have a need to know the article. Essentially, the scientist is utilizing the secure grid and the secure Semantic Web technologies not only to write the research paper but also to produce results possibly in real-time and to ensure that security is maintained at all levels. One can also envision a scenario where the secure semantic information manager essentially takes over and automates the process by invoking the secure Semantic Web as well as the secure global grid and then give pertinent results and information to the scientist depending on the access control rights (s)he has.

Secure Knowledge Management

Recently, there has been much interest in secure knowledge management (SKM, 2004). Secure knowledge management is about incorporating security into the knowledge management cycle. That is, a corporation has to protect the intellectual property and trade secrets. This is done using secure knowledge management strategies, processes, and metrics.

As we have stated earlier, knowledge management is a cyclic process. We need to investigate the security aspects. For example, when knowledge is created, one needs to ensure that the knowledge is protected. When knowledge is represented, the security policies will have to be represented also. Knowledge manipulation and dissemination involves enforcing the security policies. The actions taken need to take security into consideration.

As we have stated earlier, knowledge management components include strategies, processes, metrics, and technologies. Secure knowledge management strategies are about setting up plans for secure knowledge management. Secure processes are the actual means of sharing and enhancing the knowledge in accordance with the security policies. Metrics are about measuring the knowledge such as the number of patents obtained by a corporation. One needs to investigate the security impact on gathering metrics. Finally, technologies are the tools used for knowledge management such as the security of the Semantic Web.

Essentially, secure Semantic Web can be regarded as a tool for secure knowledge management. Web knowledge can be shared effectively and access control policies enforced appropriately. On the other hand, secure knowledge management, such as creating and managing ontologies securely, is an effective tool for the secure Semantic Web. Secure knowledge management can also be used to determine which resources to use for computing and, subsequently, enhance the power of the grid, at the same time, ensuring that the execution controls are enforced correctly. Similarly, secure grid can be used to enhance secure knowledge management by determining the way to share and allocate the resources securely. There is a lot to do on secure knowledge management as well as using this technology for securing the Semantic Web (Thuraisingham, 2004b).

Semantic Web Mining and Privacy Considerations

Overview

The standardized data format, the popularity of content-annotated documents, and the ambition of large scale formalization of knowledge of the Semantic Web has two consequences for Web mining. The first is that more structured information becomes available to which existing data mining methods can be applied with only minor modifications. The second is in the form of concept hierarchies in RDF but even more in the form of knowledge represented in OWL in combination with Web data for data mining. The combination of these two gives a form of closed-loop learning in which knowledge is acquired by Web mining and then used again for further learning. We discuss Semantic Web mining. Web mining has implications for security and privacy. Therefore, we discuss security and privacy considerations in this chapter.

Semantic Web Mining

Document Classification with the Semantic Web

Document classification methods for the Semantic Web are like those for the World Wide Web. Besides general features of documents, annotations can be used as additional features or to structure features. Knowledge in the form of ontologies can be used to infer additional information about documents, potentially providing a better basis for classification. For this, we may need to use ontologies. Hence, the key issue is to identify appropriate concepts that describe and identify documents. In this, it is important to make sure that irrelevant concepts will not be associated and matched, and that relevant concepts will not be discarded. In other words, it is important to insure that high precision and high recall will be preserved during concept selection for documents. To the best of our knowledge, in conventional keyword search, the connection through the use of ontologies between keywords and concepts selected from documents to be accessed for retrieval is carried out semi-automatically (Gonzalo, Verdejo, Chugur, & Cigarran, 1998; Smeaton & Rijsbergen, 1993), a process which is both subjective and labor intensive. Khan and Luo (2002) propose an automatic mechanism for the selection of these concepts. Furthermore, this concept selection mechanism includes a novel, scalable disambiguation algorithm using domain-specific ontology. This algorithm will prune irrelevant concepts while allowing relevant concepts to become associated with documents.

Document Clustering with the Semantic Web

If documents are similar to each other in content, they will be associated with the same concept in ontology. Now, we can construct a hierarchy which will represent ontologies at some extent. Hence, after constructing a hierarchy, we need to assign a concept for each node in the hierarchy. For this, we deploy two types of strategies and adopt a bottom-up concept assignment mechanism. First, for each cluster consisting of a set of documents, we assign a topic based on a modified Rocchio algorithm for topic tracking (Joachims, 1998). However, if multiple concepts are candidate for a topic, Khan et al. propose an intelligent method for arbitration (Khan & Luo, 2002). Next, to assign a concept to an interior node in the hierarchy, we use WordNet, a linguist ontology (Miller, 1995). Descendent concepts of the internal node will also be identified in WordNet. From these identified concepts and their hypernyms, we can identify a more generic concept that can be assigned as a concept for the interior node.

Data Mining for Information Extraction with the Semantic Web

In order to facilitate the annotation of document segments with respect to rules for extracting information and assuming that these rules are to be consistently assigned, techniques for learning to extract information must be developed. In this, knowledge of new ontologies can be of considerable benefit. Ontologies can support solutions for a host of different problems, including the learning of other ontologies and the assignment of ontology concepts to text (text annotation). These concepts may be assigned to either whole documents, as in the case of already described ontologies associated with Web documents, or to some smaller parts of text. In the case of the latter, researchers have been working on learning annotation rules from already annotated text. This can be viewed as a kind of information extraction, where the goal is not to fill in the database slots by extracted information, but to assign a label (slot name) to a part of text. Obtaining annotated text is not a trivial problem. Consequently, some researchers are investigating other techniques such as natural language processing or clustering in order to find text units (e.g., groups of nouns, clusters of sentences) and map them upon concepts of the existing ontology (Hotho, Staab, & Stunne, 2003).

Ontology Mapping

Because ontologies are often developed for a specific purpose, it is inevitable that ontologies similar to one another will be constructed. These similar ontologies must be unified to enable the use of knowledge based on one ontology in combination with knowledge from another. This requires mapping between the concepts, attributes, values, and relations in the two ontologies, either as a solution or as a step toward a single unified ontology. Several approaches to this problem have been explored by various researchers (Chalupsky, 2000; Doan, Madhavan, Domingos, & Halevy, 2004; McGuinness, Fikes, Rice & Wilder, 2000; Noy, & Musen, 2000; Stumme & Maedche, 2001). One line of attack is to first take information about concepts from the ontologies, then, extract additional information, for example, from Web pages.

Usage Mining with the Semantic Web

The Semantic Web opens up interesting opportunities for usage mining because ontologies and annotations can provide information about user actions and Web pages in a standardized form that enables discovery of richer and more informative patterns. For example, the annotation of products that are visited (and bought) by users adds information to customer segments and makes it possible to discover the underlying general patterns. Such patterns can be used, for example, to predict reactions to new products in response to the description of the new product. This would not have been possible if only the name, image, and price of the product had been available, and mining can be done much more effectively using a uniform ontology than from documents that describe products.

Security and Privacy Considerations

As we have stated in Thuraisingham (2003) and Thuraisingham et al. (2002), data mining has serious implications for security and privacy. Because of data mining, users can put pieces of data together and infer information that is highly sensitive or private. More recently, there is research on privacy preserving data mining (Thuraisingham, 2004c). While the techniques are showing some promise, there is a lot to be done before we can develop practical data mining techniques that also maintain security and privacy.

Recently, Semantic Web mining is making some progress. It is being realized that the information on the Web and the Semantic Web can be mined. We have discussed various aspects of Semantic Web mining, including document clustering and information extraction. Semantic Web mining can also be used to integrate the information on the Web. Because the Semantic Web is even more powerful than the Web, it may make the security and privacy problem more severe. Therefore, we need to examine the privacy preserving data mining techniques and determine their applicability for the Semantic Web. Essentially, the more powerful the technology is, the more serious consequences it will have for security and privacy. Therefore, we need to examine security and privacy at the on-set and not as an after-thought. Furthermore, we need to examine the implications of security and privacy for the semantic grids that result from data mining and information extraction.

Summary and Conclusions

In this chapter, we have discussed the semantic grid and examined security aspects. The semantic grid is essentially an integration of the grid and the Semantic Web. We first provided an overview of the Semantic Web, the grid, and also discussed integration issues. We also discussed knowledge management which is a critical technology for the Semantic Web. Next, we discussed security for the Semantic Web, the grid, secure integration, as well as secure knowledge management. We illustrated how these technologies could be integrated to form the secure semantic grid. Lastly, we discussed related topics such as Semantic Web mining and the implications for security and privacy.

Research on the semantic grid is just beginning. There is still a lot to do before we can develop a secure semantic grid. First of all, we need to examine security for the grid and the Semantic Web. We need to integrate the various technologies securely. Information extraction is a main function of the Semantic Web; therefore, we cannot forget about the security and privacy implications.

References

Atkins, D. (2002). *Cyber infrastructure.* NSF Document.

Berners-Lee, T., & Fischetti, M. (2000). *Weaving them Web.* San Francisco: Harper.

Bertino, E. (2004). Secure grid. *Proceedings ACM SACMAT.*

Bertino, E., et al. (2002). Access control for XML documents. *Data and Knowledge Engineering*.

Bertino, E., et al. (2004). Secure third party publication of XML documents. *IEEE Transactions on Knowledge and Data Engineering*.

Carminati, B., et al. (2004). Specifying security policies in RDF. *Proceedings of the DEXA Workshop*.

Chalupsky, H. (2000). Ontomorph: A translation system for symbolic knowledge. *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning* (pp. 471-482).

Doan, A., Madhavan, J., Domingos, P., & Halevy, A. (2004). Ontology matching: A machine learning approach. In S. Staab, & R. Studer (Eds.) *Handbook on ontology* (pp. 385-404). Berlin: Springer-Verlag.

Gonzalo, J., Verdejo, F., Chugur, I., & Cigarran, J. (1998). Indexing with WordNet synsets can improve text retrieval. *Proceedings of the Coling-ACL'98 Workshop: Usage of WordNet in Natural Language Processing Systems* (pp. 38-44).

Gruber, T.R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition, 5*(2), 199-220.

Hotho, A., Staab, S., & Stunne, G. (2003). Explaining text clustering results using semantic structures. *Proceedings of ECML/PKDD* (pp. 217-228).

Hovy, E. (1998). Combining and standardizing large-scale, practical ontology for machine translation and other uses. *Proceedings of the First International Conference on Language Resources and Evaluation (LREC)*.

IEEE04. (2004, January). IEEE intelligent sytems.

IEEE89. (1989).IEEE Computer society tutorial in parallel database systems.IEEE CS Press.

Joachims, T. (1998). A probabilistic analysis of the Rocchio Algorithm with TFIDF for text categorization. *Logic Journal of the IGPL*.

Khan, L., & Luo, F. (2002). Ontology construction for information selection. *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence* (pp. 122-127).

Khan, L., & McLeod, D. (2000). Audio structuring and personalized retrieval using ontology. *Proceedings of the ACM/IEEE Advances in Digital Libraries* (pp. 116-126).

Khan, L., McLeod, D., & Hovy, E. (2004). Retrieval effectiveness of ontology-based model for information selection. *The International Journal on Very Large Databases, 13*(1), 71-85.

McGuinness, D., Fikes, R., Rice, J., & Wilder, S. (2000). An environment for merging and testing large ontology. *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning* (pp. 483-493).

Miller, G. (1995). WordNet: A lexical database for English. *Communications of the ACM, 38*(11), 33-38.

Morey, D., et al. (2001). *Knowledge management.* Cambridge, MA: MIT Press.

Noy, N., & Musen, M. (2000). Prompt: Algorithm and tool for automated ontology merging and alignment. *Proceedings of the 17th National Conference on Artificial Intelligence* (pp. 450-455).

OWL Web Ontology Language Overview. (2004). Retrieved May 2, 2005, from *http://www.w3.org/TR/2004/REC-owl-features-20040210/*

Resource Description Framework (RDF) Model and Syntax Specification. (1999). Retrieved May 2, 2005, from *http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/*

SKM. (2004). *Secure knowledge management workshop*. Buffalo, NY.

Smeaton, A.F., & Rijsbergen, V. (1993). The retrieval effects of query expansion on a feedback document retrieval system. *The Computer Journal, 26*(3), 239-246.

Stumme, G., & Maedche, A. (2001). Fca-merge: Bottom-up merging of ontology. *Proceedings of the 17th International Conference on Artificial Intelligence* (pp. 225-230).

Thuraisingham, B. (1994). Security issues for federated database systems. *Computers and Security*.

Thuraisingham, B. (2002). Data mining, national security, privacy and civil Liberties. *SIGKDD Explorations*.

Thuraisingham, B., et al. (2002). Collaborative commerce. *Knowledge and Processes Management Journal*.

Thuraisingham, B. (2003). *Web data mining: Technologies and their applications to business intelligence and counter-terrorism*. Boca Raton, FL: CRC Press.

Thuraisingham, B. (2004a). Security standards for the Semantic Web. *Computer Standards and Interface Journal*.

Thuraisingham, B. (2004b). Secure knowledge management. *Proceedings of the SKM Workshop*.

Thuraisingham, B. (2004c). Privacy preserving data mining. *Journal of Database Management*.

Thuraisingham, B. (2005). *Database and applications security*. Boca Raton, FL: CRC Press.

Section II

Policy Management and Web Security

Chapter VI

Web Content Filtering

Elisa Bertino, Purdue University, USA

Elena Ferrari, University of Insubria at Como, Italy

Andrea Perego, University of Milan, Italy

Abstract

The need to filter online information in order to protect users from possible harmful content can be considered as one of the most compelling social issues derived from the transformation of the Web into a public information space. Despite that Web rating and filtering systems have been developed and made publicly available quite early, no effective approach has been established so far, due to the inadequacy of the proposed solutions. Web filtering is then a challenging research area, needing the definition and enforcement of new strategies, considering both the current limitations and the future developments of Web technologies—in particular, the upcoming Semantic Web. In this chapter, we provide an overview of how Web filtering issues have been addressed by the available systems, bringing in relief both their advantages and shortcomings, and outlining future trends. As an example of how a more accurate and flexible filtering can be enforced, we devote the second part of this chapter to describing a multi-strategy approach, of which the main characteristics are the integration of both list- and metadata-based

techniques and the adoption of sophisticated metadata schemes (e.g., conceptual hierarchies and ontologies) for describing both users' characteristics and Web pages content.

Introduction

In its general meaning, information filtering concerns processing a given amount of data in order to return only those satisfying given parameters. Although this notion precedes the birth of the Internet, the success and spread of Internet-based services, such as e-mail and the Web, resulted in the need of regulating and controlling the network traffic and preventing the access, transmission, and delivery of undesirable information.

Currently, information filtering is applied to several levels and services of the TCP/IP architecture. Two typical examples are spam and firewall filtering. The adopted strategies are various, and they grant, in most cases, an efficient and effective service. Yet, the filtering of online multimedia data (text, images, video, and audio) is still a challenging issue when the evaluation of their semantic meaning is required in order to verify whether they satisfy given requirements. The reason is that the available techniques do not allow an accurate and precise representation of multimedia content. For services like search engines, this results in a great amount of useless information returned as a result of a query. The problem is much more serious when we need to prevent users from retrieving resources with given content (e.g., because a user does not have the rights to access it or because the content is inappropriate for the requesting user). In such a case, filtering must rely on a thorough resource description in order to evaluate it correctly.

The development of the Semantic Web, along with the adoption of standards such as MPEG-7 (TCSVT, 2001) and MPEG-21 (Burnett et al., 2003), may seemingly overcome these problems in the future. Nonetheless, currently online information is unstructured or, in the best case, semi-structured, and this is not supposed to change in the next few years. Thus, we need to investigate how and to what extent the available techniques can be improved to allow an effective and accurate filtering of multimedia data.

In this chapter, we focus on filtering applied to Web resources in order to avoid possibly harmful content accessed by users. In the literature, this is usually referred to as Internet filtering, since it was the first example of information filtering on the Internet with relevant social entailments—that is, the need to protect given categories of users (e.g., children) from Web content not suitable for them. The expression is misleading since it could apply also to Internet services like spam and firewall filtering, already mentioned above. More properly, it should be called Web filtering, and so we do henceforth.

Web filtering is a rather challenging issue, the requirements of which have not been thoroughly addressed so far. More precisely, the available Web filtering systems focus on two main objectives: *protection* and *performance*. On one hand, filtering must prevent at all costs harmful information from being accessed—that is, it is preferable to block suitable content rather than allowing unsuitable content to be displayed. On the other hand, the filtering procedure must not noticeably affect the response delay time—that is, the evaluation of an access request and the returned result must be quickly performed.

Such restrictive requirements are unquestionably the most important in Web filtering since they grant effectiveness with respect to both user protection and content accessibility. Yet, this has resulted in supporting the rating of resource content and users' characteristics which is semantically poor. In most cases, resources are classified into a very small set of content categories, whereas only one user profile is provided. The reason is twofold. On one hand, as already mentioned above, the way Web information is encoded and the available indexing techniques do not allow us to accurately rate Web pages with the precision needed to filter inappropriate content. On the other hand, a rich semantic description would require high computational costs, and, consequently, the time needed to perform request evaluation would reduce information accessibility. These issues cannot be neglected, but we should not refrain from trying to improve flexibility and accuracy in rating and evaluating Web content. Such features are required in order to make filtering suitable to different user's requirements, unlike the available systems which have rather limited applications.

In the remainder of this chapter, besides providing a survey of the state-of-the-art, we propose extensions to Web filtering which aim to overcome the current drawbacks. In particular, we describe a *multi-strategy* approach, formerly developed in the framework of the EU project EUFORBIA,[1] which integrates the available techniques and focuses on the use of metadata for rating and filtering Web information.

Web Rating and Filtering Approaches

Web filtering concerns preventing users from accessing Web resources with inappropriate content. As such, it has become a pressing need for governments and institutional users because of the enormous growth of the Web during the last 10 years and the large variety of users who have access to and make use of it. In this context, even though its main aim is minors' protection from harmful online contents (e.g., pedophilia, pornography, and violence), Web filtering has been and still is considered by institutional users—for instance, firms, libraries, and schools—as a means to avoid the improper use of their network services and resources.

Web filtering entails two main issues: *rating* and *filtering*. Rating concerns the classification (and, possibly, the labeling) of Web sites content with respect to a filtering point of view, and it may be carried out manually or automatically, either by third-party organizations (*third-party rating*) or by Web site owners themselves (*self-rating*). Web site rating is not usually performed on the fly as an access request is submitted since it would dramatically increase the response time. The filtering issue is enforced by *filtering systems*, which are mechanisms able to manage access requests to Web sites, and to allow or deny access to online documents on the basis of a given set of policies, denoting which users can or cannot access which online content and the ratings associated with the requested Web resource. Filtering systems may be either client- or server-based and make use of a variety of filtering techniques.

Currently, *filtering services* are provided by ISP, ICT companies, and non-profit organizations, which rate Web sites and make various filtering systems available to users. According to the most commonly used strategy—which we refer to as the traditional strategy in what follows—Web sites are rated manually or automatically (by using, for instance, neural network-based techniques) and classified into a predefined set of categories. Service subscribers can then select which Web site categories they do not want to access. In order to simplify this task, filtering services often provide customized access to the Web, according to which some Web site categories are considered inappropriate by default for certain user categories. This principle is also adopted by some search engines (such as Google *SafeSearch*) which return only Web sites belonging to categories considered appropriate.

Another rating strategy is to attach a *label* to Web sites consisting of some metadata describing their content. This approach is adopted mainly by PICS-based filtering systems (Resnick & Miller, 1996). PICS (Platform for Internet Content Selection) is a standard of the World Wide Web Consortium (W3C) which specifies a general format for *content labels*. A PICS content label describes a Web page along one or more dimensions by means of a set of *category-value* pairs, referred to as *ratings*. PICS does not specify any labeling vocabulary: this task is carried out by *rating* or *labeling services*, which can specify their own set of PICS-compliant ratings. The filtering task is enforced by tools running on the client machine, often implemented in the Web browser (e.g., both Microsoft Internet Explorer and Netscape Navigator support such filters).[2]

Compared to the category models used in the traditional strategy, PICS-based rating systems are semantically richer. For instance, the most commonly used and sophisticated PICS-compliant rating system, developed by ICRA (Internet Content Rating Association: www.icra.org), provides 45 ratings, grouped into five different macro-categories: *chat*, *language*, *nudity and sexual material*, *other topics* (promotion of tobacco, alcohol, drugs and weapons use, gambling, etc.), *violence*.[3] On the other side, the category model used by RuleSpace EATK™, a tool adopted by Yahoo and other ISPs for their parental control services, makes use of 31 Web site categories.[4]

On the basis of the adopted filtering strategies, filtering systems can then be classified into two groups: *indirect* and *direct filtering*. According to the former strategy, filtering is performed by evaluating Web site ratings stored in repositories. These systems are mainly based on *white* and *black lists*, specifying sets of good and bad Web sites, identified by their URLs. It is the approach adopted by traditional filtering systems. The same principle is enforced by the services known as *walled gardens*, according to which the filtering system allows users to navigate *only* through a collection of preselected good Web sites. Rating is carried out according to the third-party rating approach.

Direct filtering is performed by evaluating Web pages with respect to their actual content, or the metadata associated with them. These systems use two different technologies. *Keyword blocking* prevents sites that contain any of a list of banned words from being accessed by users. Keyword blocking is unanimously considered the most ineffective content-based technique, and it is currently provided as a tool which can be enabled or disabled by user's discretion. *PICS-based filtering* verifies whether an access to a Web page can be granted or not by evaluating (a) the content description provided in the PICS label possibly associated with the Web page and (b) the filtering policies specified by the user or a supervisor. PICS-based filtering services adopt a self-rating approach, usually providing an online form which allows Web site owners to automatically generate a PICS label.

According to analyses carried out by experts during the last years,[5] both indirect and direct filtering techniques have several drawbacks, which can be summarized by the fact that they *over*- or *under*-block. Moreover, both category models adopted by traditional rating services and PICS-based rating systems have been criticized for being too Western-centric so that their services are not suitable for users with a different cultural background. Those analyses make clear that each filtering technique may be suitable (only) for certain categories of users. For instance, though white and black lists grant a very limited access to the Web, which is not suitable for all users, these techniques, and especially walled garden-based services, are considered as the safest for children.

The available PICS-based rating and filtering services share similar drawbacks. The content description they provide is semantically poor: for instance, none of them makes use of ontologies, which would allow a more accurate content description. Moreover, it is limited to content domains considered liable to be filtered according to the Western system of values. Nevertheless, the PICS standard can be considered, among the available technologies, the one which can better address the filtering issues. Since its release in 1996, several improvements have been proposed, first of all, the definition of a formal language for PICS-compliant filtering rules, referred to as *PICSRules* (W3C, 1997). Such a language makes the task of specifying filtering policies according to user profiles easier, and it could be employed by user agents to automatically tailor the navigation of users. Finally, in 2000, the W3C proposed an RDF implementation of PICS (W3C, 2000), which provides a more expressive description of Web sites content, thus enabling more sophisticated filtering.

Despite these advantages, the PICS-based approach is the less diffused. The reason is that it requires Web sites to be associated with content labels, but currently, only a very small part of the Web is rated. PICS-based services, like ICRA, have made several efforts to establish the practice of self-rating among content providers, but no relevant results have been obtained. Moreover, both the PICS extensions mentioned above (PICSRules and PICS/RDF) did not go beyond the stage of proposal. Nonetheless, the wider and wider use of XML and its semantic extensions—that is, RDF (W3C, 2004b) and OWL (W3C, 2004a)—make metadata-based approaches the major research issue in the filtering domain. Consequently, any improvement in Web filtering must take into consideration both the current limitations and the future developments of Web technologies.

So far, we have considered how the rating and filtering issues have been addressed with respect to the *object* of a request (Web sites). Yet, which content is appropriate or inappropriate for a user depends on his/her characteristics—that is, access to a given Web site can be granted or prevented depending on the requesting user. Consequently, filtering may be considered as a process entailing the evaluation of both subjects (users) and objects (Web sites) properties. The simplest case is when users share the same characteristics: since only a single *user profile* is supported, the filtering parameters are set by default, and there is no need to evaluate the characteristics of a single user. This applies also when we have one or more *predefined* user profiles or when filtering policies concern specific users. Such an approach, which we refer to as *static user profiling*, is the one adopted by the available filtering systems, and it has the advantage of simplifying the evaluation procedure (only Web sites characteristics must be evaluated), which reduces the computational costs. On the other hand, it does not allow flexibility; thus, it is not suitable in domains where such a feature is required.

Following, we describe a filtering approach which aims at improving, extending, and making more flexible the available techniques by enforcing two main principles: (a) support should be provided to the different rating and filtering techniques (both indirect and direct), so they can be used individually or in combination according to users' needs and (b) users' characteristics must be described accurately in order to provide a more effective and flexible filtering.

Starting from these principles, we have defined a formal model for Web filtering. The objective we pursued was to design a general filtering framework, addressing both the flexibility and protection issues, which can possibly be customized according to users' needs by using only a subset of its features.

Multi-Strategy Web Filtering

In our model, users and Web pages (resources, in the following) are considered as entities involved in a communication process, characterized by a set of properties and denoted by an identifier (i.e., a URI). A filtering policy is a rule, stating which users can/cannot access which resources. The users and resources to which a policy applies are denoted either explicitly by specifying their identifiers, or implicitly by specifying constraints on their properties (e.g., "all the users whose age is less than 16 cannot access resources with pornographic content"). Note that the two types of user/resource specifications (explicit and implicit) are abstractions of the adopted filtering strategies: explicit specifications correspond to list-based approaches (i.e., white/black lists and walled gardens); implicit specifications merge all the strategies based on ratings (e.g., PICS) and/or content categories (e.g., RuleSpace).

Users' and resources properties are represented by using one or more *rating systems*. Since rating systems are organized into different data structures, and support should be provided to more complex and semantically rich ones (e.g., ontologies), we represent them as sets of ratings, hierarchically organized and characterized by a set, possibly empty, of attributes. That is to say, we model rating systems as ontologies. This approach has two advantages. The uniform representation of rating systems allows us to enforce the same evaluation procedure. Thus, we can virtually support any current and future rating system without the need of modifying the model. Moreover, the hierarchical structure into which ratings are organized allows us to exploit a *policy propagation* principle according to which a policy concerning a rating applies also to its children. This feature allows us to reduce as much as possible the policies to be specified, keeping their expressive power.

Our model is rather complex considering the performance requirements of Web filtering. Its feasibility must then be verified by defining and testing strategies for optimizing the evaluation procedure and reducing as much as possible the response time. We addressed these issues during the implementation of the model into a prototype, the first version of which was the outcome of the EU project EUFORBIA (Bertino, Ferrari, & Perego, 2003), and the results have been quite encouraging. The current prototype greatly improves the computational costs and the performance of the former ones. Thanks to this, the average response delay time is now reduced to less than 1 second, which does not perceivably affect online content accessibility.

Following, we describe the main components of the model and the architecture of the implemented prototype, outlining the strategies adopted to address performance issues.

The MFM Filtering Model

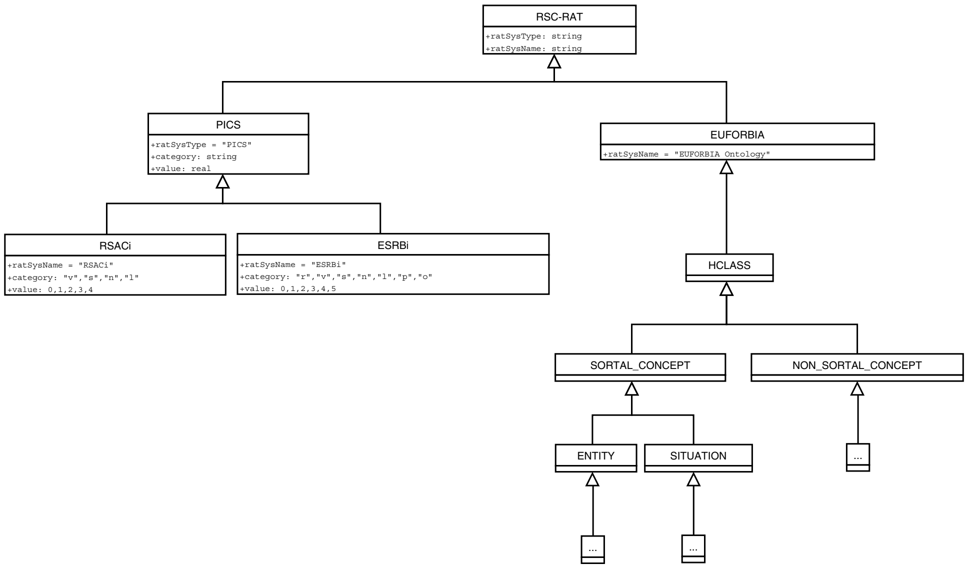
In our model, referred to as MFM (*Multi-strategy Filtering Model*), we use the notion of *agent* to denote both the active and passive entities (users and resources, respectively) involved in a communication process, the outcome of which depends on a set of *filtering policies*. An agent *ag* is a pair (*ag_id*, *ag_pr*), where *ag_id* is the agent identifier (i.e., a URI), and *ag_pr* are the agent *properties*. More precisely, *ag_pr* is a set, possibly empty, of *ratings* from one or more *rating systems*. Users and resources are then formally represented according to the general notion of agent by the pairs (*usr_id*, *usr_pr*) and (*rsc_id*, *rsc_pr*), respectively.

As mentioned above, MFM rating systems are modeled into a uniform structure—that is, as a set of ratings hierarchically organized. Moreover, the whole set of rating systems is structured into two *super-trees*, one for rating systems applying to users and one for rating systems applying to resources. Each super-tree has a root node at level 0 of the hierarchy, which is the parent of the root node of each rating system. As a result, the several rating systems possibly supported are represented as two single rating systems, one for users and one for resources. This *extrinsic* rating system integration totally differs from approaches as the ABC-based one (Lagoze & Hunter, 2001) aiming to provide semantic interoperability among ontologies. Yet, all the attempts to define a rating *meta*-scheme, representing the concepts commonly used in the available rating systems, have been unsuccessful so far. Our objective is then to harmonize only their structure in order to easily specify policies ranging over different rating systems.

Figure 1 depicts an example of a resource rating system super-tree, whose root is RSC-RAT, merging two PICS-based rating systems (namely, the RSAC*i* and ESRB*i* ones[6]) and the conceptual hierarchy developed in the framework of the project EUFORBIA. For clarity's sake, in Figure 1, only the upper levels of the tree are reproduced.

Ratings are denoted by an identifier and characterized by a set, possibly empty, of attributes and/or a set of *attribute-value* pairs. We then represent a rating *rat* as a tuple (*rat_id*, *attr_def*, *attr_val*), where *rat_id* is the rating identifier (always a URI), *attr_def* is a set, possibly empty, of pairs (*attr_name*, *attr_domain*), defining the name and domain of the corresponding attributes, and *attr_val* is a set, possibly empty, of pairs (*attr_name*, *value*).

Figure 1. Example of a resource rating system super-tree

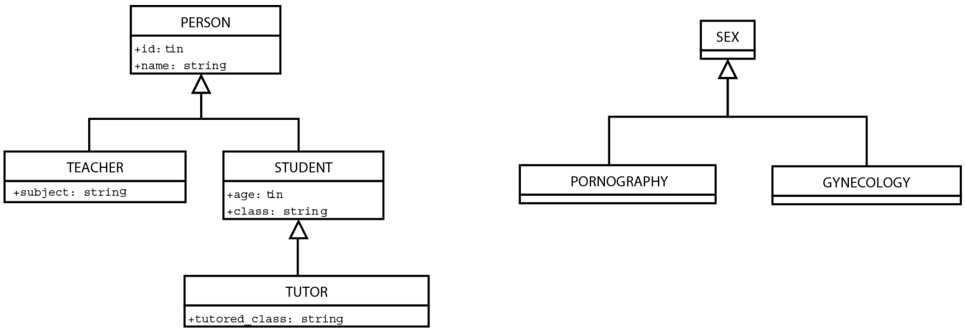


Attributes and attribute-value pairs are ruled according to the object-oriented approach. Thus, (a) attributes and attribute-value pairs specified in a rating are inherited by its children, and (b) attributes and attribute-value pairs redefined in a child rating override those specified in the parents. Examples of ratings are those depicted in Figure 1.

The last MFM key notion to consider before describing filtering policies is that of *agent specification*. To make our examples clear, following, we refer to Figure 2, depicting two simple user (a) and resource (b) rating systems.

In MFM, we can denote a class of agents by listing them explicitly (*explicit agent specifications*) or specifying constraints on their properties (*implicit agent specifications*). Implicit agent specifications are defined by a *constraint specification language* (CSL), which may be regarded as a Description Logic (Baader et al., 2002) providing the following constructs: concept intersection, concept union, and comparison of concrete values (Horrocks & Sattler, 2001). In CSL, rating systems are then considered as IS-A hierarchies over which concept operations are specified in order to identify the set of users/resources satisfying given conditions.

Figure 2. Examples of user (a) and resource (b) rating systems



(a) *(b)*

Examples of CSL expressions referring to the rating system in Figure 2(a) may be TEACHER ⊔ TUTOR (i.e., "all the users associated with a rating TEACHER or TUTOR") and STUDENT`.age > 14` (i.e., "all the users associated with a rating STUDENT and whose age is greater than 14"). Note that thanks to the rating hierarchy, a CSL expression concerning a rating applies also to its children. For instance, the CSL expression STUDENT`.age > 14` denotes also the users associated with a rating TUTOR, provided that their age is greater than 14.

A filtering policy is then a tuple (*usr_spec*, *rsc_spec*, *sign*), where *usr_spec* is an (explicit or implicit) agent specification denoting a class of users (*user specification*); *rsc_spec* is an (explicit or implicit) agent specification denoting a class of resources (*resource specification*); and *sign* ∈ {+,–} states whether the users denoted by *usr_spec* can (+) or cannot (–) access resources denoted by *rsc_spec*.

The sign of a policy allows us to specify exceptions with respect to the propagation principle illustrated above. Exceptions are ruled by a conflict resolution mechanism according to which, between two conflicting policies (i.e., policies applying to the same user and the same resource but with different sign), the prevailing is the more specific one.

Finally, when the rating hierarchies cannot be used to solve the conflict, negative policies are considered prevalent. This may happen when the user/resource specifications in two policies are equivalent or when they denote disjoint sets of ratings, which are yet associated with the same users/resources.

Example 1.

Let us suppose that no user can access contents regarding the sexual domain, unless he/she is a teacher or a tutor. Moreover, we allow students whose age is greater than 14 to access contents regarding gynecology. These requirements can be enforced by specifying the following validations: pv = (PERSON, SEX, –), fp_2 = (TEACHER ⊔ TUTOR, SEX, +), and fp_3 = (STUDENT`.age > 14`, GYNECOLOGY, +). It is easy to realize that policies fp_2 and fp_3 are in conflict with fp_1. Nonetheless, according to our conflict resolution mechanism, fp_2 is more specific than fp_1 since TEACHER and TUTOR are children of PERSON, whereas fp_3 is more specific than fp_1 since STUDENT is a child of PERSON, and GYNECOLOGY is a child of SEX. As a consequence, fp_2 and fp_3 prevail over fp_1. Consider now a 15-year-old user, whose identifier is `Bob`,[7] associated with a rating `rat1` instance of STUDENT, and a policy fp_4 = ({`Bob`}, GYNECOLOGY, –): fp_4 prevails over fp_3 since the user specification is explicit and therefore is more specific. The same principle applies to resources. Thus, given a Web site `www.example.org`, associated with a child of the rating SEX, a policy fp_5 = (PERSON, {`www.example.org`}, +) prevails over fp_1.

The propagation principle and the conflict resolution mechanism are also applied to resources by exploiting the URI hierarchical structure (IETF, 1998). Thus, a policy applying to a given Web page applies as well to all the resources sharing the same URI upper components (e.g., a policy applying to `www.example.org` applies as well to `www.example.org/examples/`). In case of conflicting policies, the stronger is the one concerning the nearer resource with respect to the URI syntax.

As demonstrated by the example above, our approach allows us to reduce, as much as possible, the set of policies which need to be specified providing high expressive power and flexibility. Moreover, MFM is fully compliant with RDF and OWL. This implies that we can use RDF/OWL as standard cross-platform encoding for importing/exporting data structured according to our model among systems supporting these technologies.

Supervised Filtering

In MFM, policies are specified only by the System Administrator (SA), and no mechanism to delegate access permissions is supported. Yet, in some domains, the responsibility of specifying filtering policies may be shared among several persons, and in some cases, the opinions of some of them should prevail. For instance, in a school context, teachers' opinions should prevail over the SA's, and parents' opinions should prevail over both the teachers' and the SA's. Moreover, it could be often the case that certain categories of users (e.g., minors) should be subject to a very restrictive access to the Web according to which they can access a given Web page only if their supervisors agree.

In order to satisfy these requirements, MFM also supports the notion of *supervised filtering*, according to which a given set of users in the system, referred to as *supervisors* (*SV* for short), should validate the filtering policies specified for a given set of users, referred to as *supervised users* (*SU* for short), are valid or not before they can be effective. Additionally, sometimes a kind of supervision is required according to which access request made by a user must be approved by a supervisor, even though it is authorized by a filtering policy. Different sets of supervisors may correspond to different sets of supervised users, and supervisors cannot belong to the corresponding set of supervised users.

Supervised users are grouped into two disjoint subsets, denoted by two different *supervision modes*, *normal* and *strict*. Depending on the supervision mode associated with a supervised user, supervised filtering is enforced either *implicitly* or *explicitly*, respectively. In the normal supervision mode, a filtering policy applies to a supervised user only if the corresponding supervisors agree. In the strict supervision mode, a supervised user can access a specific resource only if the corresponding supervisors agree. Thus, filtering policies applying to users associated with a normal supervision mode must be validated by the authorized supervisors. On the other hand, access requests to Web pages submitted by users associated with a strict supervision mode must be authorized by a filtering policy and *explicitly* by supervisors.

The supervised filtering component of MFM (MFM-SF) is modeled, as the base one, according to an agent-oriented approach: *active* agents are *supervisors,* and *passive* agents are *supervised users*. Users with the role of supervisor are denoted by associating *supervisor ratings* with them, hierarchically structured into one or more *supervisor rating systems*, which are in turn grouped into a super-tree. We can then denote, either explicitly or implicitly, a class of supervisors by means of a *supervisor specification*—that is, an agent specification concerning supervisors. Similarly, a class of supervised users is denoted by a *supervised user specification*—namely, a user specification applying only to the set of supervised users. If we denote by *US* and *SUS* the sets of user and supervised user specifications, respectively, it follows that $SUS \subseteq US$. Finally, a set of supervised users and the corresponding set of supervisors are denoted by a *supervision specification*, which is a pair (*sv_spec*, *su_spec*), where *sv_spec* and *su_spec* are, respectively, a supervisor and a supervised user specification.

In this context, the rating hierarchy is used to denote the authority of a supervisor according to the following principles: (a) child ratings have a stronger authority than their parents, whereas (b) ratings at the same level of the hierarchy have the same authority. For instance, let us suppose two supervisor ratings PARENT and TEACHER, such that PARENT $\prec$ TEACHER: we say that supervisors associated with the rating PARENT have stronger authority than those associated with the rating TEACHER.

We can now introduce the notions of *filtering policy validation* (*validation*, for short) and *supervised filtering policy*. A validation *fpv* is a tuple (*sv_id*, *su_spec*, *FP*, *sign*), where *sv_id* is the identifier of the supervisor who validated the set of policies *FP* for the users denoted by the supervised user specification *su_spec*, whereas $sign \in \{+,-\}$ states whether the policy is valid (+) or not (–). Similarly, a supervised filtering policy *sfp* is a tuple (*sv_id*, *usr_spec*, *RI*, *sign*), where *sv_id* is the identifier of the supervisor who decided that supervised users denoted by *su_spec* can or cannot, depending on the value of $sign \in \{+,-\}$, access resources whose identifier is in *RI*.

Example 2

Let us suppose two users `Jane` and `Ted`, who are, respectively, `Bob`'s mother and teacher—consequently, `Jane` and `Ted` are associated with supervisor ratings PARENT and TEACHER, respectively. We can denote `Jane` and `Ted` as supervisors of `Bob` by an explicit supervision specification ({`Bob`}, {`Jane`, `Ted`}). We can also state that teachers are supervisors of students by an implicit supervision specification (STUDENT, TEACHER). Let us now suppose that Ted does not agree with fp_3—that is, he does not agree that 15-aged students can access content concerning gynecology. On the contrary, Jane agrees with fp_3, as far as her son Bob is concerned. Ted and Jane state their decisions by specifying the following validations: fpv_1 = (`Ted`, STUDENT, $\{fp_3\}$, –) and fpv_2 = (`Jane`, {`Bob`}, $\{fp_3\}$, +).

In the example above, we then have two conflicting validations. In order to identify the prevailing, we check the authority degree of each supervisor. Since `Jane` is a parent and `Ted` is a teacher, according to the supervisor rating hierarchy, fpv_2 prevails over fpv_1. If supervisors' authority is equally strong, in order to solve a possible conflict between validations, we compare the specificity degree of their supervised user specification components. For instance, given a teacher `Mary` and a validation fpv_3 = (`Mary`, STUDENT`.age = 15`, {fp_3}, +), fpv_3 prevails over fpv_1 since it is more specific with respect to the supervised user specification. In case the two supervised user specifications are equally specific, the prevailing validation is the negative one.

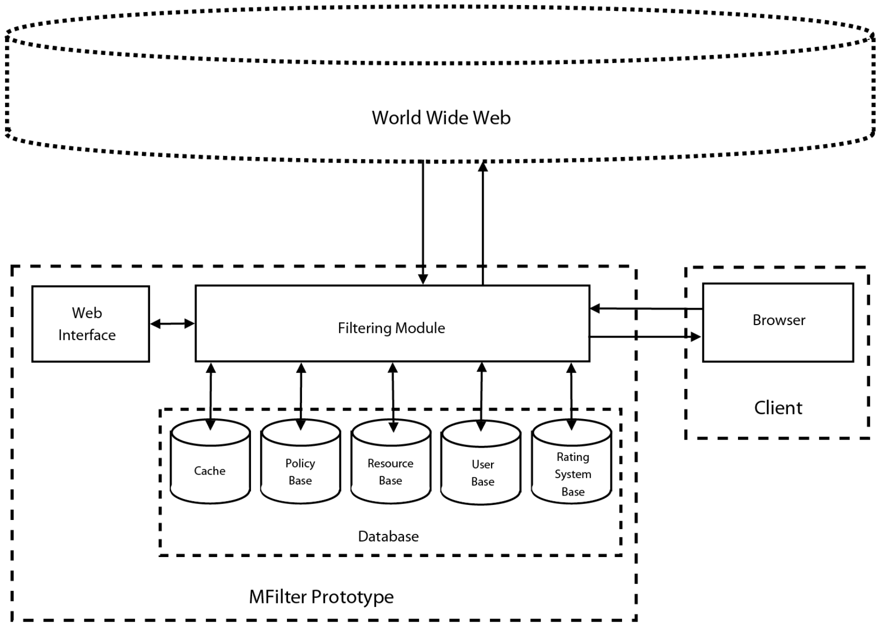
Finally, let us now consider the following examples of supervised filtering policies: sfp_1 = (`Ted`, {`Bob`}, {`www.example.org`}, –), sfp_2 = (`Jane`, {`Bob`}, {`www.example.org`}, +), and sfp_3 = (`Mary`, STUDENT, {`www.example.org`}, +). In case of conflict, the prevailing supervised filtering policy is identified according to the same principles illustrated above for filtering policies and validations. Thus, sfp_2 prevails over sfp_1 since Jane has a stronger authority than Ted, whereas sfp_1 prevails over sfp_3 since the supervised user specification in the former policy is more specific than the one in the latter.

Model Implementation: Architecture and Performance Issues

Figure 3 depicts the architecture of the implemented prototype, referred to as MFilter. MFilter is a Java-based system built on top of the Oracle DBMS, and it is structured into three main components. The first is the Filtering Module, which intercepts each access request submitted by users and verifies whether it can be granted or not according to the filtering policies specified by the SA and stored in the MFilter Database. The second is the Database, which stores all the information needed by the system to perform the filtering tasks. The third is the Web Interface, structured into three main components (the *Administration*, the *Supervision*, and the *User Interfaces*), which allow the management of the system, user authentication, and the specification/validation of policies.

The core of the system is the Filtering Module (FM), which evaluates all the submitted access requests and returns the requested Web documents or an access denied message according to the filtering policies specified for the requesting user. The evaluation procedure is quite a complex task, dealing with several issues. Following, we focus on the main one: *performance*.

Figure 3. MFilter architecture



Given an access request (*usr_id*, *rsc_id*), the FM computes the *policy projection* concerning the requesting user and the requested resource—that is, it retrieves all the policies applying to user with identifier *usr_id* and resource with identifier *rsc_id*—and determines the strongest one. Policy projection and evaluation have high computational costs which, if carried out at runtime, would dramatically increase the delay time between the access request and the system response. In order to improve the efficiency and effectiveness of the evaluation procedure and to reduce its computational costs at runtime, the FM enforces precomputational strategies and caching mechanisms, which are carried out by different submodules. Whenever a policy is specified or updated, the system computes all the users to whom the policy applies. Moreover, the system stores the URIs and metadata possibly associated with all the Web pages requested by the users. This allows us to precompute all the already requested resources to which a policy applies and, consequently, to greatly reduce the response delay time if a request concerns one of these Web pages. As a consequence, since a given set of users access mainly a precise subset of the Web, the best performance is reached as soon as the system has stored the information concerning the usually accessed Web pages. Thus, system performance improves the more intensely the system is used.

Web Filtering and Access Control

In this chapter, we defined and discussed Web filtering only with respect to the need of preventing users from accessing inappropriate Web contents. Yet, the same techniques can be used whenever access to a resource can be granted only if given constraints, concerning either the end user or the resource, are satisfied. More precisely, Web filtering shares several similarities with access control. Although in the latter, *resources*, not *users*, are the entities to be protected, this does not necessarily affect how filtering/access control policies are formalized. Actually, a filtering policy is not different from an authorization: it simply states that a given set of users can/cannot access a given set of resources.

This is particularly true for MFM. Our model supports notions—such as negative/positive policies, policy propagation, and strong/weak policies—available in some discretionary access control models. The main difference concerns access *privileges*: MFM allows one to state only whether a resource can or cannot be viewed, whereas access control usually supports privileges concerning the modification of resource content (*write*, *update*, *append*, etc.). Yet, such privileges can be easily added in MFM by extending the notion of filtering policy and representing it by a tuple (*usr_spec*, *rsc_spec*, *priv*, *sign*), where *priv* denotes the access privilege. Moreover, we can organize privileges into a hierarchy in order to resolve conflicts: for instance, we can say that a write privilege is *stronger* than a read privilege; consequently, given two filtering policies *fp* = (PERSON, {`www.example.org`}, `write`, +) and *fp2* = (PERSON, {`www.example.org`}, `read`, –), *fp* is considered prevalent since it is stronger with respect to the access privilege.

Nonetheless, the possibility of specifying policies based on multiple rating systems and the support for supervision make MFM more flexible than traditional access control approaches. Such features are not provided by the available access control models but in a limited form. *Credentials* describing users' characteristics and policies based on the content of resources rely on predefined attribute/category sets, and supervision is usually not supported, unless in the rather different principle of *administration delegation*. As a result, MFM is particularly suitable to enforce access control/filtering in Web and distributed environments, where the characteristics of users and resources may be described using different metadata vocabularies and where the policy specification task is not necessarily centralized.

Conclusions

Metadata-based filtering is today one of the most promising approaches to a controlled distribution of Web contents. By contrast, list- or keyword-based strategies have proved to enforce a very restrictive (and often rather ineffective) filtering, which, bounding its application to a very narrow set of user categories, has been one of the major reasons of the missed diffusion of these services among Web users. Web filtering *per se* is rather a sensitive problem which has been and is still debated by governments, industry, and libertarian institutions. The most diffused and well-grounded criticism is that it may enforce institutional censorship in online free speech. As a matter of fact, PICS aimed at overcoming this criticism by delegating and users or supervisors (e.g., parents) the responsibility of deciding which content is appropriate or inappropriate. The attempt has been unsuccessful since PICS-based rating systems provide a content classification which, besides being semantically poor, is constrained to what is considered liable to be filtered. As a consequence, a PICS rating system is already a judgment on what is inappropriate according to a given cultural background. Then, the PICS approach, in principle, is still valid, but it must provide a semantically richer rating of Web content, not constrained to a particular value system.

We moved in this direction in the framework of the EUFORBIA project, funded by the EU Safer Internet Action Plan (`www.saferinternet.org`) by defining a general-purpose rating system (namely, the EUFORBIA conceptual hierarchy) which allows one to accurately describe Web sites structure and content. EUFORBIA labels may be compared with the metadata scheme of a *digital item* (DI) in MPEG-21 (ISO, 2003), although MPEG-21 focuses mainly on the problem of describing the structure and the access rights of a DI. Such approach, currently not adopted by Web rating and filtering systems, will seemingly become the standard way to represent online content as the recent semantic extensions to MPEG-21 can testify (see Hunter, 2003). The possibility of enforcing an effective and flexible filtering, where end-users can freely decide what they wish and what they do not wish to access, may probably solve the ethical disputes concerning Web filtering and may also reduce end user resistance in adopting them (Ying Ho & Man Lui, 2003), removing eventually two of the main obstacles to its diffusion.

The multi-strategy filtering model illustrated in this chapter is an attempt to address the drawbacks of the existing tools by integrating and optimizing the available technologies and to define a framework compliant with the upcoming Web technologies. In particular, the recent efforts of the W3C in defining a standard architecture for Web services (W3C, 2004c), which will incorporate Web content filtering into a broader framework, are quite relevant, concerning the evaluation of online information with respect to given user requirements. Starting from this, we are now investigating the feasibility of our approach in other filtering-related areas, more precisely that concerning the quality of service, where access to online resources is granted or prevented depending on whether the quality policies declared by the service provider satisfy those defined by users in their profile.

References

Baader, F., Calvanese, D., McGuinness, D., Nardi, D., & Patel-Schneider, P. (Eds.) (2002). *The description logic handbook: Theory, implementation, and applications*. Cambridge: Cambridge University Press.

Bertino, E., Ferrari, E., & Perego, A. (2003, November). Content-based filtering of Web documents: The MaX system and the EUFORBIA project. *International Journal of Information Security*, *2*(1), 45-58.

Burnett, I., Van de Walle, R., Hill, K., Bormans, J., & Pereira, F. (2003, October-December). MPEG-21: Goals and achievements. *IEEE Multimedia, 10*(4), 60-70.

Cranor, L.F., Resnick, P., & Gallo, D. (1998, September). Technology inventory: A catalog of tools that support parents' ability to choose online content appropriate for their children. Retrieved May 2, 2005, from *http://www.research.att.com/projects/tech4kids*

Horrocks, I., & Sattler, U. (2001). Ontology reasoning in the SHOQ(D) description logic. In B. Nebel (Ed.), *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI 2001)* (pp. 199-204). Morgan Kaufmann.

Hunter, J. (2003, January). Enhancing the semantic interoperability of multimedia through a core ontology. *IEEE Transactions on Circuits and Systems for Video Technology, 13*(1), 49-58.

IETF. (1998, August). Uniform Resource Identifiers (URI): Generic syntax. Retrieved May 2, 2005, from *http://www.ietf.org/rfc/rfc2396.txt*

Infopeople Project. (2001, May). Internet filtering options analysis: An interim report. Retrieved May 2, 2005, from *http://www.infopeople.org/howto/filtering/InternetFilter_Rev1.pdf*

ISO. (2003). *Information technology—Multimedia framework (MPEG-21)—Part 2: Digital item declaration*. Geneve: International Organization for Standardization.

Lagoze, C., & Hunter, J. (2001, November 6). The ABC ontology and model. *Journal of Digital Information.* Retrieved May 2, 2005, from *http://jodi.ecs.soton.ac.uk/Articles/v02/i02/Lagoze*

Neumann, P.G., & Weinstein, L. (1999). Inside risks: Risks of content filtering. *Communications of the ACM, 42*(11), 152.

Resnick, P., & Miller, J. (1996, October). PICS: Internet access controls without censorship. *Communications of the ACM, 39*(10), 87-93.

Sobel, D. (Ed.). (2001). *Filters & freedom 2.0: Free speech perspectives on Internet content controls*. Washington, DC: Electronic Privacy Information Center.

TCSVT. (2001, June). *IEEE Transactions on Circuits and Systems for Video Technology—Special Issue on MPEG-7, 11*(6).

W3C. (1997, December 29). PICSRules 1.1. Retrieved May 2, 2005, from *http://www.w3.org/TR/REC-PICSRules*

W3C. (2000, March 27). PICS rating vocabularies in XML/RDF. Retrieved May 2, 2005, from *http://www.w3.org/TR/rdf-pics*

W3C. (2004a, February 10). *OWL Web ontology language: Overview.* Retrieved May 2, 2005, from *http://www.w3.org/TR/owl-features*

W3C. (2004b, February 10). *Resource description framework (RDF): Concepts and abstract syntax.* Retrieved May 2, 2005, from *http://www.w3.org/TR/rdf-concepts*

W3C. (2004c, February 10). *Web services architecture.* Retrieved May 2, 2005, from *http://www.w3.org/TR/ws-arch*

Weinberg, J. (1997, Winter). Rating the net. *Hasting Communications and Entertainment Law Journal, 19*(2), 453-482.

Ying Ho, S., & Man Lui, S. (2003). Exploring the factors affecting Internet content filters acceptance. *ACM SIG e-Com Exchange, 4*(1), 29-36.

Endnotes

[1] The EUFORBIA project Web site is available at http://e-msha.msh-paris.fr/Agora/ Tableaux%20de%20bord/Euforbia/

[2] The complete PICS reference documentation is available online at http://www.w3.org/PICS

[3] The list of ICRA ratings is available at http://www.icra.org/faq/decode

[4] The list of RuleSpace categories is available at http://www.rulespace.com/products/models.php

[5] See, for instance, Weinberg (1997), Cranor, Resnick, and Gallo (1998), Neumann and Weinstein (1999), Infopeople Project (2001), Sobel (2001).

[6] We have chosen these rating systems since their simple structure is more suitable for examples. RSAC*i* (Recreational Software Advisory Council on the internet: www.rsac.org) is the predecessor of ICRA, and it is no longer available; ESRB*i* (Entertainment Software Rating Board Interactive: www.esrb.org) is a non-profit organization providing ratings for the entertainment software industry.

7 Following, for clarity's sake, we adopt symbolic identifiers for users instead of URIs.

Chapter VII

Sanitization and Anonymization of Document Repositories

Yücel Saygin, Sabanci University, Turkey

Dilek Hakkani-Tür, AT&T Labs—Research, USA

Gökhan Tür, AT&T Labs—Research, USA

Abstract

Information security and privacy in the context of the World Wide Web (WWW) are important issues that are still being investigated. However, most of the present research is dealing with access control and authentication-based trust. Especially with the popularity of WWW as one of the largest information sources, privacy of individuals is now as important as the security of information. In this chapter, our focus is text, which is probably the most frequently seen data type in the WWW. Our aim is to highlight the possible threats to privacy that exist due to the availability of document repositories and sophisticated tools to browse *and analyze these documents. We first identify possible threats to privacy in document repositories. We then discuss a measure for privacy in documents with some possible solutions to avoid or, at least, alleviate these threats.*

Introduction

Information has been published in various forms throughout the history, and sharing information has been one of the key aspects of development. The Internet revolution and World Wide Web (WWW) made publishing and accessing information much easier than it used to be. However, widespread data collection and publishing efforts on the WWW increased the privacy concerns since most of the gathered data contain private information. Privacy of individuals on the WWW may be jeopardized via search engines and browsers or sophisticated text mining tools that can dig through mountains of Web pages. Privacy concerns need to be addressed since they may hinder data collection efforts and reduce the number of publicly available databases that are extremely important for research purposes such as in machine learning, data mining, information extraction/retrieval, and natural language processing.

In this chapter, we consider the privacy issues that may originate from publishing data on the WWW. Since text is one of the most frequently and conveniently used medium in the WWW to convey information, our main focus will be text documents. We basically tackle the privacy problem in two phases. The first phase, referred to as *sanitization*, aims to protect the privacy of the contents of the text against possible threats. Sanitization basically deals with the automatic identification of named entities such as sensitive terms, phrases, proper names, and numeric values (e.g., credit card numbers) in a given text, and modification of them with the purpose of hiding private information. The second phase, called *anonymization*, makes sure that the classification tools cannot predict the owner or author of the text.

In the following sections, we first provide the taxonomy of possible threats. In addition to that, we propose a privacy metric for document databases based on the notion of *k*-anonymity together with a discussion of the methods that can be used for preserving privacy.

Background and Related Work

Privacy and security issues were investigated in the database community in the context of statistical databases, where the users are limited to statistical queries. In statistical databases, privacy is protected by limiting the queries that can be issued by the user to non-confidential values, or statistical operations (Adam & Wortmann, 2004). Security leaks resulting from the intersection of multiple queries are investigated, and privacy is defined by the concept of *k*-anonymity. A database table provides *k*-anonymity if it cannot be used to unambiguously identify less than k entities (Samarati & Sweeney, 1998).

Currently, the data mining community is investigating how data could be mined without actually seeing the confidential values. This is called privacy preserving data mining which was introduced in Agrawal and Srikant (2000) for the case of classification model construction. Further research results have been published on various data mining models for preserving privacy (Evfimievski, Srikant, Agrawal, & Gehrke, 2002; Rizvi & Haritsa, 2002). Privacy preserving data mining on distributed data sources was another interesting research direction, which was addressed in Clifton and Kantarcioglu (2004) and Vaidya and Clifton (2004) for association rule mining and classification model construction. Another aspect of privacy issues in data mining is to protect the data against data mining algorithms. This result is due to the fact that data mining tools can be used to discover sensitive information. Hiding sensitive association rules by database sanitization is proposed in Saygin et al. (2001) and Verykios et al. (2004). Further research was conducted for data sanitization to protect the data against data mining tools (Oliveira & Zaïane, 2002, 2003). However, there is not much work about preserving privacy for natural language databases and its effects, except the studies of Ruch et al. (2000) and Sweeney (1996) who have worked on sanitization of medical reports on a limited domain.

On the other hand, information extraction (IE) has been studied extensively in the natural language processing community. IE is the task of extracting particular types of entities, relations, or events from natural language text or speech. The notion of what constitutes information extraction has been heavily influenced by the *Message Understanding Conferences* (MUCs). The basic information extraction task of *Named Entity Extraction* covers marking names—and determining their types as persons, locations, or organizations—and certain structured expressions—money, percent, date, and time (Hakkani-Tur et al., 1999). An example text from the Broadcast News, whose named entities are marked, is given in Figure 1. These entities can also be marked using

Figure 1. An example text from Broadcast News corpus. Named entities are marked: bold for dates, italics for locations, underlines for organizations and gray for person names

…The other very big story of the **today** is in *Washington* where the <u>White House</u> administration has already been badly shaken up by the possibility that president Clinton and one of his advisors Vernon Jordan obstructed justice…

XML tags. It may also be useful to make HTML links of entities and co-referencing pronomials, which point to pages where summary information about the entity such as the gender of a person or further references in the text, can be listed.

A task related to information extraction is the automatic authorship identification, which aims to assign authors to texts. This topic is studied mainly in linguistics, computational linguistics, and stylometrics fields beginning in the precomputer era (Oakes, 1998). Commonly used features include vocabulary of the author, usage of some typical words (mostly stopwords) and phrases, and construction style of sentences (e.g., average sentence length, part of speech tags in the first words of the sentences, most common voice, etc.). In our previous work, we have found that using just the word (unigram) frequencies, it is possible to detect the author of a newspaper article by more than 90% accuracy when the candidate set is fixed to nine authors and using about 100,000 words per author for training (Tur, 2000).

Privacy Threats in Documents

Privacy issues in document repositories arise from the fact that the text contains private information, which can be jeopardized by the adversaries who are curious to know more about individuals for various reasons. Adversaries could turn this information into advantage, such as the private information that may be published in a tabloid.

Since our main concern is document repositories, the main elements we are going to consider are documents as information sources. There are also what we call Privacy Conscious Entities (PCEs) whose privacy may be jeopardized

Table 1. Relationships among privacy threats, private information, and tools for extracting private information

Type of private information	**Tools that can be used by the adversary**	**Type of threat**	
Explicit	*Browsers, Editors*	*Direct*	
Implicit	*Record Matching, Information Retrieval*	*Via Data Integration*	*Indirect*
	Data Mining Statistical Analysis	*Via Data Analysis*	

by the release of the documents. A PCE could be a person, company, or organization. Privacy concerns of PCEs may require that the identifying information of the PCE (person name, company name, etc.) should not be seen in a document since the document is related with a sensitive topic. A PCE may not want to be seen as the author of the document or appear in a document in a certain context, such as being a criminal or being in debt. A PCE may not want to be associated with another entity, such as being a friend of a convict. In doing so, the links between the documents, such as references or hyperlinks to a document, should also be taken into account.

The private information in a text could be grouped into two classes, namely, explicit and implicit information. Explicit information could be the name, salary, or address of a person that could be viewed by text editors, browsers, and can be found by search engines. Implicit information could be the characteristics of a document such as the author, the statistical properties like the frequencies of some words and punctuation marks, or usage of particular phrases that can be identified with an individual. Data mining and statistics tools are needed to reach such information.

In Table 1, we listed the type of private information that could be obtained by an adversary, the tools that could be used for the corresponding private information type, and the type of threat. As shown in Table 1, we classify the privacy threats as direct and indirect. Direct threats occur due to the existence and availability of explicit information in the data such as the name of the person including some extra information regarding the salary of that person. Names of individuals, phone numbers, and salary amounts are just a few examples forming a direct threat to privacy when they are revealed to a third party. Upon the disclosure of the text data, users can see the contents using a browser or an editor. Indirect threats can be of two kinds: one is due to data integration, and the other is caused by data analysis. The former type of indirect threat is the integration of different documents in order to infer private information that cannot be revealed by each individual document when considered alone. The integration can be achieved by finding those documents that are considered "similar" based on the similarity measures defined in the context of information retrieval (Cohen, 2000). Indirect threats via data analysis, instead, are due to the application of machine learning and data mining techniques over the available data. New data mining tools, especially for text classification, can be used with a training database (which is easy to construct from news groups, etc.) to infer private information such as the author of a document.

Preserving Privacy in Documents

In this section, we first propose a measure for privacy in document repositories using the notion of *k*-anonymity. We then address the privacy problem in two phases. The former phase is called *sanitization*. It deals with the automatic extraction of named entities such as sensitive terms, phrases, proper names, and numeric values from a given text. Extracted terms are then replaced with dummy values, or more generic terms depending on the privacy requirements. The latter phase, known as *anonymization*, makes sure that the classification tools cannot predict the owner or author of the text. We should note that all the sanitization and anonymization techniques can be applied to spoken language as well. For example, in the case of anonymization, the purpose may be to hide the identity of the speaker.

k-Anonymity as a Privacy Measure in Documents

Privacy in documents can be assessed using the notion of k-anonymity that has been heavily investigated in the context of statistical disclosure control (Samarati & Sweeney, 1998). *k*-anonymity is related to the identification of entities (individuals, companies, etc.) in released data where the confidential attributes are hidden [15, 19]. For example, the data in a hospital regarding patients that includes the patient name, surname, social security number (SSN), postal code (ZIP), birth date, sex, diagnosis, and the medication should not be disclosed without removing the name, surname, and SSN columns which identify a person. However, when we remove the name, surname, and SSN, one may think that ZIP, birth date, sex, diagnosis, and the medication can be safely released without the risk of privacy violation. It turns out that this is not the case because when combined with publicly available data such as a voter list, we may recover the SSN, name, and surname information from the voter list database using the ZIP, birth date, and sex columns from the released data. In tabular data sources, a set of attributes (such as the ZIP, birth date, and sex) is called *quasi-identifier* if it could be used in connection with public data to identify a person (Sweeney, 1996). Quasi-identifiers are used as a base for measuring the anonymity provided by the released data with respect to publicly available data sources. The degree of anonymity is called *k*-anonymity in general, and it is formally defined in Sweeney (1996) for tabular data as follows:

Definition (k-anonymity). Let $T(A_1, A_2, \ldots, A_n)$ be a table with attributes A_1 through A_n, and QI be a quasi-identifier associated with it. T is said to satisfy k-anonymity with respect to QI if and only if each sequence of values in T[QI] appears at least with k occurrences in T[QI] where T[QI] denotes the projection on attributes QI maintaining duplicates.

k-anonymity property makes sure that the disclosed data cannot be used to distinguish a person among a group of k or more individuals. In relational data model, which was heavily adopted by statistical database researchers, the data is structured; therefore, the attribute values related to an individual are in the same row clearly identifying their relationship to the individual. However, in case of unstructured text, a major issue is to find the terms that identify an individual, or that are related to an individual. Similar to the definition in Sweeney (2002), we define the set of named entities that relate to the same individual and that can be used to identify an entity in relation to other publicly available text as *quasi-identifier named entities*. Quasi-identifier named entities can be date, sex, address, and so on. We need to make sure that these terms cannot be used to differentiate between a set of entities, where the set size is k. In case of authorship detection, we need to make sure that the author of a document cannot be identified among less than k-authors to provide *k*-anonymity. We define *k*-anonymity in case of the authorship of a document as follows:

Definition (k-anonymity of authorship). Let D_P be a set of documents whose authors are known, D_C be a set of documents whose authorship is confidential, and A be the set of authors of the documents in $D_P U D_C$. A document d_i D_C satisfies k-anonymity of authorship with respect to D_P, if D_P can not be used to form a prediction model that will reduce the set of possible authors to A_P where $A_P \subseteq A$, and $|A_P| < k$.

Text Sanitization

The aim of sanitization is to protect the privacy of individuals given their privacy requirements. The first step of the sanitization process is to extract personally identifying information such as the name, SSN of a person, or company name if we would like to protect the privacy of a company. However, spotting the personally identifying information may not be enough; we also need to find the quasi-identifier named entities that could be used to identify individuals by linking to other documents such as the ZIP, birth date, and gender.

Sanitization depends on the corresponding task. There are three known methods for partial access to databases (Conway & Strip, 1976), which can also be used for sanitization:

1. Value distortion alters the confidential values to be hidden with random values.
2. Value dissociation keeps these values but dissociates them from their actual occurrence. This can be achieved, for example, by exchanging the values across sentences.
3. Value-class membership exchanges the individual values with disjoint, mutually exhaustive classes. For example, all the proper names can be changed to a single token Name.

The simplest form of sanitization is modifying the values of named entities or replacing them with generic tokens as in the value-class membership approach. If the named entities are not already marked using XML tags, we can utilize automatic named entity extraction methods, which are well studied in the computational linguistics community. The concept of *k*-anonymity can be assured for text sanitization while determining the generic tokens. For example, people names can be generalized until they map to at least k-people. For the case of numeric values such as salary, a concept hierarchy can be exploited. The salary can be mapped to a more generic value, which refers to at least k people in a specific context. (e.g., low, average, high, and astronomic linguistic hedges in the concept hierarchy) even when quasi-identifier information is used. For the case of addresses, we can ensure that the address maps to k different people in the company or a district where at least k distinct addresses exist.

The generic tokens can also preserve the non-sensitive information to ensure readability of the text. For example, the gender or identification of the people can be marked in the token for the resolution of further (pronominal) references (i.e., <PERSON> versus <PERSON, GENDER=MALE>). An even harder task would be associating references during sanitization as in the example below where <DATE2> is extended as <DATE2=DATE1+3 days>. A sample text and its sanitized version is provided in Figure 2.

Another example is the task of automatic call classification, which is an emerging technology for automating call centers. During the development of call routing systems, previous customer calls are recorded and then transcribed. A human annotator examines the transcriptions and assigns them a call-type from a set of predefined call-types. This data is then used to train automatic speech recognition and call classification components. Figure 3 presents an example dialog between an automated call center agent and a user. As it is clear from this example, these calls may include very sensitive information, such as credit card and phone numbers, that needs to be sanitized before this data can be shared or made public.

Figure 2. A modified example text from a medical record (Tur et al., 1999) and its sanitized version

Dear Dr. Blue,
Your patient, Mr. John Brown, stayed in our service from 05/05/1999 to 05/08/1999.
Mr. Brown, 72 year old, has been admitted to emergency on 05/05/1999. His tests for the cytomegalovirus and the EBV were negative. Therefore, Dr. George Green performed an abdominal CT scan. Mr. Brown will be followed in ambulatory by Dr. Green…

Dear Dr. <PER1>,
Your patient, Mr. <PER2>, stayed in our service from <DATE1> to <DATE2=DATE1+3>.
Mr. <PER2>, 72 year old, has been admitted to emergency on <DATE1>. His tests for the cytomegalovirus and the EBV were negative. Therefore, Dr. <PER3> performed an abdominal CT scan. Mr. <PER2> will be followed in ambulatory by Dr. <PER3>…

Figure 3. Example dialog from a automatic call center recording and its sanitized version

System: How may I help you?
User: Hello. This is John Smith. My phone number is 9 7 3 area code 1 2 3 9 6 8 4. I wish to have my bill, long distance bill, sent to my Discover card for payment. My Discover card number is 2 8 7 4 3 6 1 7 8 9 1 2 5 7 hundred and it expires on first month of next year.

System: How may I help you?
User: Hello. This is <NAME>. My phone number is <PHONE_NUMBER>. I wish to have my bill, long distance bill, sent to my <CREDIT_CARD> for payment. My <CREDIT_CARD> number is <CREDIT_CARD_NUMBER> and it expires on <DATE>.

One problem with text sanitization is that the performance of the state-of-the-art information extraction techniques is still far from being perfect (especially when employed for previously unseen text or domains). In order not to miss any confidential information, one may choose high recall for low precision, which may end up with more numbers of falsely sanitized portions of text. With the value-class membership approach, the missed named entities will be easy to recognize. Thus, if the named entity extraction performance is low, using value distortion or dissociation methods would be more appropriate. Another problem is the domain dependency of the confidential information. For example, some named entities may be confidential for only some domains, such as drug names in the medical reports vs. pharmaceutical company customer care center recordings, requiring a list of the entities that should be sanitized.

Text Anonymization

Text anonymization aims at preventing the identification of the author (who is also considered to be the owner) of a given text. In the case of speech, the speaker is considered to be the owner. Text anonymization is therefore necessary to protect the privacy of the authors. For example, the identity of the reviewers of scientific papers would prefer to be anonymous. This is also the case for authors of the papers in blind reviews.

We know that, using a state-of-the-art classifier, it is possible to identify the author of a text with a very high accuracy. The features that can be used are the words and phrases (n-grams) in the text, the total number of tokens, total number of different word types, total number of characters, and the number of word types that occur once. We have identified through our experiments with a limited set of articles from a newspaper that each author uses a characteristic frequency distribution over words and phrases. We use *k*-anonymity of authorship as the privacy metric for anonymization that was defined in the *k*-Anonymity as a Privacy Measure in Documents section. For the anonymization process, we may assume a fixed set of documents such as a digital library which collects all the major works of a given set of authors. In this case, authorship information for some documents are known, and some of them are not known. However, we should also consider the case when the adversary is able to find another set of documents for the authors, for example, by searching the Internet where the number of documents that could be found is practically infinite.

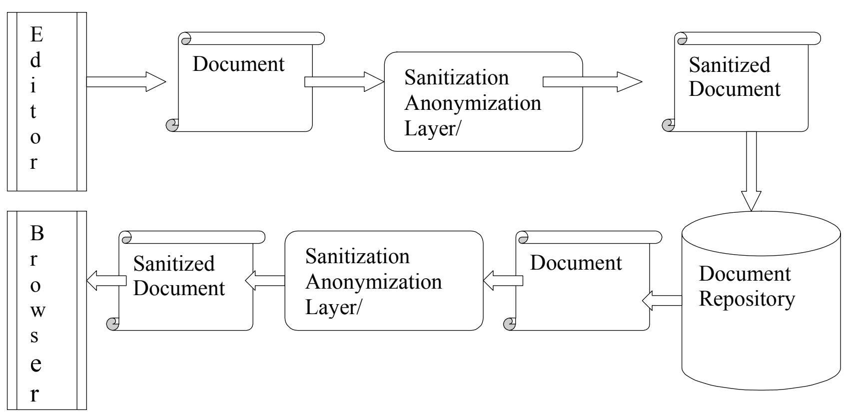
Text classification techniques first parse the text to obtain the features. Each document is represented as a feature vector where each feature may be represented by a real number. In case of a fixed document set, let D_P be the set of documents where the authorship information is public and D_A be the set of documents where the authorship information is confidential. An adversary could train a classification model using D_P to predict the authorship information of a document in D_A. Since D_P is known and fixed, anonymization can work on both D_P and D_A. The basic approach for anonymization is to modify the documents in D_P and D_A in order to change their feature vectors so that the data mining tools cannot classify the document accurately. The most general model that an adversary may use is a classification model that returns probabilities $P(a_j|d_i)$ for each author a_j for a given document d_i. In this way, each author will have a certain probability of being an author for a specific anonymous document. The basic approach that can be used for achieving *k*-anonymity is to change the probability of the real author so that (s)he falls into one of top 1…k positions randomly selected among the top-k authors with highest probability. Probabilities are then changed by updating the documents in D_P and D_A. This process should be performed in such a way that the original meaning and coherence of the document is preserved. When D_P is not fixed, then the model that could be constructed by the adversary cannot be known in advance, which complicates the anonymization process. In this case, the approach would be to update the anonymous documents in such a way that their feature vectors look alike to obscure the adversary. We can achieve this by changing the feature vectors such that at least k of the documents with different authors have the same feature vector which can be done by taking the mean of k feature vectors of documents with different authors and assigning the mean as the new feature vector.

The anonymization method heavily depends on the features of the classifier used for authorship identification by the adversary. If the classifier only uses unigram word distributions, then anonymization can be achieved simply by changing the words with their synonyms or by mapping them to more generic terms as done by sanitization. If the classifier uses a different feature set, such as the distribution of the stop-words (*the* or *by*) or words from closed class part of speech (word category) tags (that is almost all words which are not noun, verb, or adjective), then revising the sentences would be a solution as in text watermarking (Atallah et al., 2002). If the classifier uses other features such as passive or active voice, specific clauses, average length of sentences, and so forth, they need to be addressed specifically. If the text anonymization task has no information about the features of the classifier adversary used, then the optimal solution would be to assume that it uses all the features we can think of and anonymize the text accordingly.

Discussion of a System for Preserving Privacy in Document Repositories

A system for anonymization and sanitization is depicted in Figure 4. As can be seen in the figure, sanitization and anonymization can be viewed as a layer between the mediums of interaction with the user and document repository. Users may create a document using an editor, and upon their request, the document may be sanitized before it is stored. The same process works in the

Figure 4. Sanitization and anonymization for documents



opposite direction as well. When a user wants to view a document, the document could be sanitized (if it is stored in its original form in a trusted document repository) before it could be viewed.

Data Quality Measures for Text

We need to make sure that the data quality is still preserved after the sanitization. Data quality could be measured in terms of readability and the ability to use the sanitized text for the corresponding task. For example, if the data is going to be used for text classification, it is necessary to perform sanitization without deteriorating the classification performance. If the task is information retrieval, sanitization methods should not interfere with the indexing and document matching methods.

Information hiding aims at inserting additional information into any kind of digital media (Katzenbeisser & Petitcolas, 2000). This information can be intended to be a concealed message to be read only by specific people and not by other parties (steganography), a code identifying/protecting the owner (watermarking), or a code identifying the receiver (fingerprinting). The availability of media in digital form made the unauthorized copying and distribution of these very easy, increasing the concerns and therefore research for protecting the copyright. One recent study on text watermarking digitizes the sentence using its syntactic parse tree, embeds the hidden information into the tree by changing its structure, and regenerates a new sentence with the same meaning from this structure (Atallah et al., 2002). The subject of this chapter is not to insert hidden information into text, instead it is to hide the sensitive information in the text without distorting its meaning. However, evaluation measures can be shared across information hiding, sanitization, and anonymization tasks since all have the requirement that they should not change the meaning and the coherence of the original text during the update process. Possible information theoretic measures for data quality are Kullback Leiblar distance and change in conditional entropies, the details of which details can be found in Cover and Thomas (1991).

Future Trends

Standards for privacy policies and privacy preference specifications are being developed under the W3C with the Platform for Privacy Preferences (P3P) Project (http://www.w3.org/P3P/). A method for implementing the P3P standard was proposed using database technology by Agrawal, Kiernan, Srikant, and Xu (2003). As a future research direction, sanitization and anonymization tools should consider the privacy preferences of users and privacy policies of data collectors. Another important aspect is the development of online techniques for sanitization and anonymization. This is becoming more important especially with the emerging portals with online e-mail scanning capabilities.

Threats that occur due to data integration from multiple sources need further investigation. Simply preprocessing the data may not be enough to ensure privacy. Data integration and record linkage can be used to identify individuals from data sources sanitized by different mechanisms and different policies. For example, the same type of text collected from different sources may be released in sanitized form. However, one may sanitize the names, and one may sanitize sensitive data values due to inconsistent sanitization policies. Standardization of sanitization mechanisms and policies for specific data types is needed for ensuring privacy in large-scale distributed document repositories.

Conclusions

In this chapter, we identified the privacy issues in document repositories and pointed out some approaches to tackle the privacy protection problem. The ideas we presented aim to identify the problem and propose some initial solutions to it combining existing technology from natural language processing and data mining. The initial ideas we presented will hopefully lead to more research in this direction and the development of tools between the users of documents and the storage medium that will ensure the privacy requirements. With the privacy issues in text identified, tools for protecting the privacy can be developed which will lead to the release of more text data without the need of money and time consuming text preprocessing done by humans. In sum, text sanitization and anonymization will ensure privacy of individuals and serve to increase the richness of data sources on the Web.

References

Adam, N.R., & Wortmann, J.C. (1989). Security-control methods for statistical databases: A comparative study. *ACM Computing Surveys, 21*(4), 515-556.

Agrawal, R., Kiernan, J., Srikant, R., & Xu, Y. (2003). Implementing P3P using database technology. *Proceedings of the 19th International Conference on Data Engineering,* March, Bangalore, India.

Agrawal, R., & Srikant, R. (2000). Privacy preserving data mining. *Proceedings of SIGMOD Conference* (pp. 45-52).

Atallah, M., Raskin, V., Hempelmann, C., Karahan, M., Sion, R., Topkara, U., & Triezenberg, K. (2002). Natural language watermarking and tamperproofing. *Information Hiding,* 96-212.

Clifton, C., & Kantarcioglu, M. (2004). Privacy-preserving distributed mining of association rules on horizontally partitioned data. *IEEE Transactions on Knowledge and Data Engineering, 16*(9).

Cohen, W. (2000). Data integration using similarity joins and a word-based information representation language. *ACM Transactions on Information Systems, 18*(3), 288-321.

Conway, R., & Strip, D. (1976). Selective partial access to a database. *Proceedings of the ACM Annual Conference*.

Cover, M., & Thomas A. (1991). *Elements of information theory*. New York: John Wiley & Sons.

Evfimievski, A., Srikant, R., Agrawal, R., & Gehrke, J. (2002). Privacy preserving mining of association rules. *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 217-228).

Hakkani-Tür, D., Tur, G., Stolcke, A., & Shriberg, E. (1999). Combining words and prosody for information extraction from speech. *Proceedings of the EUROSPEECH'99, 6th European Conference on Speech Communication and Technology*, Budapest, Hungary.

Katzenbeisser, S., & Petitcolas, F. (Ed.) (2000). *Information hiding techniques for steganography and digital watermarking.* Norwood, MA: Artech House.

Oakes, M. (1998). *Statistics for corpus linguistics*. Edinburgh University Press.

Oliveira, S., & Zaïane, O. (2002). Privacy Preserving Frequent Itemset Mining. *Proceedings of the IEEE ICDM Workshop on Privacy, Security and Data Mining* (pp. 43-54).

Oliveira, S., & Zaïane, O. (2003). Protecting sensitive knowledge by data sanitization. *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM'03)* (pp. 613-616).

Rizvi, S., & Haritsa, J. (2002). Privacy-preserving association rule mining. *Proceedings of 28th International Conference on Very Large Data Bases*.

Ruch, P., Baud, R.H., Rassinoux, A.M., Bouillon, P., & Robert, G. (2000). Medical document anonymization with a semantic lexicon. *Journal of American Medical Informatics Association*.

Samarati, P., & Sweeney, L. (1998). *Protecting privacy when disclosing information: k-Anonymity and its enforcement through generalization and suppression.* SRI-CSL-98-04, Computer Science Library, SRI International.

Saygin, Y., Verykios, V.S., & Clifton, C. (2001). Using unknowns to prevent the discovery of association rules. *SIGMOD Record, 30*(4), 45-54.

Sweeney, L. (1996). Replacing personally-identifying information in medical records, the Scrub System. *Proceedings of the American Medical Informatics Association Meeting* (pp. 333-337).

Sweeney, L. (2002). k-Anonymity: A model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 10*(5), 557-570.

Tur, G. (2000). *Automatic authorship identification* (Technical Report). Retrieved May 3, 2005, from *http://www.research.att.com/~gtur/pubs.html*

Vaidya, J., & Clifton, C. (2004). Privacy preserving Naïve Bayes Classifier for vertically partitioned data. *Proceedings of the 2004 SIAM Conference on Data Mining,* Lake Buena Vista, FL.

Verykios, V.S., Elmagarmid, A., Bertino, E., Saygin, Y., & Dasseni, E. (2004). Association rule hiding. *IEEE Transactions on Knowledge and Data Engineering, 16*(4), 434-447.

Chapter VIII

Integrating Access Policies into the Development Process of Hypermedia Web Systems

Paloma Díaz, Universidad Carlos III de Madrid, Spain

Daniel Sanz, Universidad Carlos III de Madrid, Spain

Susana Montero, Universidad Carlos III de Madrid, Spain

Ignacio Aedo, Universidad Carlos III de Madrid, Spain

Abstract

This chapter discusses the integration of access control in the development process of hypermedia applications. Two key ideas are proposed: the use of high level, abstract access control models and the inclusion of access control in the whole life cycle of hypermedia applications. Authors present an access control model for hypermedia that makes it possible to formalize access policies using elements of the hypermedia domain, those used to specify structure and navigation services. Abstract models are not enough

to assist developers in dealing with security in a systematic way. Thus, authors describe how high-level access rules can be specified following the Ariadne Development Method (ADM). The ARCE project is used as example of development.

Introduction

The hypermedia paradigm organizes information as an associative net of nodes that can be freely browsed by selecting links and using navigation tools such as indexes, breadcrumbs, or maps. From the point of view of the application, Web sites are a subclass of hypermedia systems that use a specific technology to manage and deploy information but share the same access philosophy, and, for that reason, we will use the term hypermedia Web system to denote a special case of hypermedia. Access control is an essential requirement in hypermedia Web systems. Most companies and organizations are taking profit from the distributed nature of the Web to provide advanced services to authorized users (e.g., financial transactions). During the development process of hypermedia Web systems, access requirements have to be tackled from different levels of abstraction (Fernández et al., 1996) and not just as implementation decisions. At the highest level of abstraction, the focus of this chapter, access models provide formal mechanisms to determine who can or cannot do what with which components of the hypermedia Web system, that is, to specify the access rules, both the positive as well as the negative, that establish a safe system operation. In fact, a security model for hypermedia/Web allows formalization of access policies using components and services that belong to the hypermedia domain, the same as those used for the specification of other system features (i.e., structure and navigation services). Thus, developers will be able to decide and discuss issues such as which hypermedia nodes can be accessed, which information items are to be delivered in each node, or which links are to be made available according to users' permissions, without being aware of how and where hypermedia components are physically stored. It is important to note that high-level models are not enough to assist developers in dealing with security in the systematic way that distinguishes engineering from other disciplines. Access modeling has to be integrated into the whole development process (Devanbu & Stubblebine, 2000), so that, from the beginning of the project, developers know how to specify access policies and how to relate access models with the rest of the design products in order to avoid conflicts or inconsistencies among requirements. This chapter deals with the integration of access policies into the software development life cycle of hypermedia Web systems. In particular, it describes how hypermedia/Web developers can specify high-level access rules following a systematic approach using the Ariadne Development Method (ADM).

Background

The next section reviews three basic issues concerning the goals and motivations of the work described in this chapter: the special features of the development process of hypermedia/Web systems; the integration of access policies modeling into the whole development process, and some existing hypermedia access models.

The Development Process of Hypermedia Web Systems

Hypermedia development poses rather specific problems for which specific modeling mechanisms are required (Garzotto, Paolini, & Schwbe, 1993; Nanard & Nanard, 1995). First, and concerning the features of the development process itself, it is much more incremental and iterative than in the rest of software systems (Lowe & Webby, 1999). Another point is that hypermedia is interaction, and any interaction model has to be evaluated to demonstrate its practical utility. Therefore, hypermedia development requires not only a multidisciplinary team, including software engineers, programmers, or multimedia developers but also an active involvement of stakeholders (e.g., clients, potential users, etc.). The involvement of stakeholders in the development process not only encompasses the adoption of a user-centered approach and the evaluation of prototypes; stakeholders also have to take part in the validation of design models. With this purpose, hypermedia methods have to provide notations understandable by a heterogeneous audience, not necessarily interested in technological details. Second, and concerning the design models generated during the development process, software engineering methods, including object-oriented, lack elements and mechanisms to gather aesthetic and cognitive aspects of hypermedia design (Nanard & Nanard, 1995). In particular, six design views have to be considered (Garzotto et al., 1993; Díaz, Aedo, & Montero, 2001): (1) *Navigation Design*, since the hypermedia browsing structure and the navigation tools added to avoid user disorientation constitute a critical concern in hypermedia design; (2) *Presentation Design*, as nodes include multimedia contents that need to be organized and harmonized in different dimensions (like space and time) and that have to be useful and usable at the same time; (3) *Structure Design*, since most hypermedia systems have an intrinsic hierarchical structure that has to be gathered by means of data models; (4) *Behaviour Design,* as hypermedia systems are highly interactive and include aspects like the system reaction to events, access to external applications (like databases or inference engines), or the inclusion of virtual objects and structures created or modified at runtime; (5) *Processes Design*, because besides navigation functionalities, hypermedia systems are including more and more non-navigation functions such as business processes, search engines, or personalization and customization functionalities; and (6) *Access Design*, as the need to preserve due security and adapt the system to the user needs and preferences is increasing. Consequently, specific mechanisms based on abstractions of the hypermedia domain (such as nodes, links, navigation tools, or space and time-based relationships) and knowledge on hypermedia practices such as the use of conditional navigation tools (Isakowitz, Kamis, & Koufaris, 1998), contextual navigation (Schwabe & Rossi, 1998), or event-based specifications (Díaz, Aedo, & Panetsos, 2001), are required to produce high quality hypermedia Web systems. A comparative study of existing hypermedia engineering methods can be found in Montero, Díaz, and Aedo (2003).

The Integration of Access Modeling into the Software Development Process

Integrating access control modeling into the whole development process makes it possible to produce a well-documented and safe system, where security requirements are unified with other requirements and are easy to test, maintain, and reuse. In fact, such an integration is being addressed by the Federal Agency Security Practices that counts on a specific area of best practices for "Integrating security into the systems development life cycle" (Grance, Hash, & Stevens, 2003). When such a *holistic* engineering approach is not adopted, security requirements are often added to the system once it has been implemented. Such a process tends to be costly and error-prone, since access policies have to be shoehorned into existing code instead of being developed at the same time. Integration can help in considering smarter solutions in the implementation phase (e.g., use of design patterns or aspect-oriented programming). Moreover, programmers are forced to avoid conflicts with other functional and non-functional requirements already implemented (Devanbu & Stubblebine, 2000). For instance, an access constraint can somehow compromise efficiency or consistency issues; therefore, the relationship among different design views has to be maintained in order to be aware of collateral effects. Finally, the integration of security requirements in analysis and design tasks also helps developers and stakeholders to become aware of the preeminent role security plays in their systems. Developers tend to brandish two arguments to forget security requirements until the very end of the project: it is costly and time-consuming, and hypermedia Web systems have to be deployed in a rush. However, if security requirements are not properly considered, the resulting system can be useless. As pointed out in Brose, Koch, and Löhr (2001), the difficulty of security analysis and modeling can be overcome when security models are integrated with the rest of the system models. For instance, works like Devanbu and Stubblebine (2000) claim for UML extensions to unify security with other requirements, provided that this modeling technique has become a *de facto* industrial standard in object-oriented software development. Such a challenge has been taken up in works like Jürgens (2002) and Brose et al. (2001). The authors of this chapter consider that the integration of access modeling into the development process of hypermedia/Web systems is a most relevant concern: not only access control is treated as a key requirement, but also access constraints are unified with other hypermedia requirements such as structural, navigation, presentation, interaction, or behavior needs. In addition to avoiding conflicts, this integration approach allows the design and implementation of smart solutions.

Hypermedia Access Models

The need for hypermedia-specific access control models is widely recognized due to the special characteristics of hypermedia models and applications, which make the adaptation of access control models for DBMS or OO systems that do not support hypermedia components and services difficult and inappropriate. The implications of these issues when defining hypermedia authorization models are thoroughly discussed in Bertino and Samarati (1995). Earlier hypermedia formal models, including Campbell and Goodman (1988) and Stotts and Furuta (1989), used *discretionary access control* (DAC) mechanisms oriented toward ensuring confidentiality, but they cannot be used to control which kind of operations users can perform once they have been granted access. There are also high-level security models for hypermedia which allow the definition of security rules, although they are not concerned with the specification of other hypermedia features. In this way, the multi-level security model proposed in Thuraisingham (1995) applies a MAC mechanism based on classifying objects and subjects using the typical confidentiality levels of military applications (unclassified, secret, and top-secret). Instead of making use of such privacy levels, security levels can be defined on the basis of the manipulation abilities users can have, as in Samarati, Bertino, and Jajodia (1996), allowing designers to establish who can do what (e.g., activating a link, modifying contents). This last work considers several hypermedia features such as fine-grained access or the protection of links. Also, a basic composition mechanism between nodes is defined based on directories. Another approach is the use of *role-based access control* (RBAC) mechanisms, where the authorization rules are defined in terms of roles and operations. As Joshi, Aref, Ghafoor, and Spafford (2001) point out, the RBAC paradigm is an attractive candidate for Web-based applications due to its potential support for multi-domain environments. The first attempt to integrate RBAC mechanisms in the Web is reported by Ferraiolo, Barkley, and Kuhn (1999) as a continuation of previous works started in 1997. This work describes in detail the NIST RBAC model that became a standard recently (ANSI, 2004), as well as the RBAC/Web implementation for Web servers within an intranet. This proposal focuses on Web technology, so no relation with abstract hypermedia concepts (such as links, structural, navigational, and operational features) is provided. Objects are viewed as a whole with no relationships between them; therefore, there is no possibility of considering fine-grained access. In Bertino et al. (2000), the distributed nature of the Web leads authors to extend the subject concept, supporting several approaches, including user groups, RBAC, or credential-based access control. In addition, the authorization representation and enforcement can also vary in order to better meet the needs derived from decentralized architectures (access control lists vs. capabilities). Finally, the concept of domain is adopted to simplify the administration of authorizations when objects reside in different hosts. Recent works on access control for XML, whether using an RBAC approach (Zhang, Park, & Sandhu, 2003) or not (Damiani et al., 2002), are also relevant, though they focus on a specific technology while access models for hypermedia provide abstract concepts (like node, link, content, etc.) that can be mapped into different technological options, including XML. In this context, one step forward is represented by access control for Web services that are intended to achieve interoperability in distributed, heterogeneous environments. Here, access control needs to be extended in order to reflect new requirements such as temporal and context-dependent constraints or user credentials. The MARAH access model for hypermedia and Web (Aedo, Díaz, & Montero, 2003), the one on which the ADM is based, assumes some of these features, but it is strongly based on the data conceptualization principle, so access rules are specified in terms of abstract hypermedia components.

ADM Development Method

The Ariadne Development Method is a software engineering method establishing a process to specify and evaluate hypermedia and Web systems (Díaz, Aedo, & Montero, 2001; Díaz, Montero, & Aedo, 2004). The ADM is a method devoted to the development of large- and medium-scale hypermedia/ Web systems that specifies a complete, systematic, iterative, flexible, and user-centered development process counting on three phases (see Figure 1): *Conceptual* and *Detailed Design* address requirements design from different abstraction levels and the *Evaluation* phase that is based on prototypes and design models assessment. The method does not impose any sequence (arrows in the figure represent relationships among phases), so developers can follow a top-down, bottom-up, or any flexible approach. Each phase is decomposed into some activities, in each of which various models are produced (Table 1).

Figure 1. The ADM process

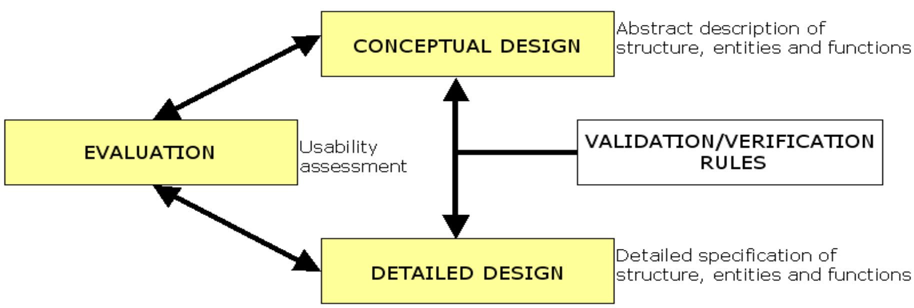


Table 1. ADM phases, activities, and models

CONCEPTUAL DESIGN		
Activity	**Models**	**Concern**
Definition of the logical structure	Structural Diagram	Structural relationships among nodes (information containers)
Study of the system function	Navigation Diagram	Navigation paths and tools
	Functional Specifications	Non navigation services
Specification of entities	Internal Diagrams Spatial Diagram Timeline	 Node visualization area Node evolution throughout a time interval
	Attributes Catalogue	Properties and metadata
	Events Catalogue	Behaviours
Users Modelling	Users Diagram	Roles or stereotypes used for security or personalization purposes and that can have attributes
Definition of the security policy	Categorization Catalogue	Security category for each hypermedia object
	Authorization Rules	Permissions that determine the operations users can perform
DETAILED DESIGN		
Activity	**Models**	**Concern**
Identification of instances	Diagram of Nodes Instances	Instances of the abstract elements or structures
	Instanced Users Diagram	Instances of user structures
Specification of functions	Specifications of Access Structures	Detailed description of navigation aids
	Detailed Specification of Functions	Detailed description of each function
Specification of instances	Detailed Internal Diagrams	Detailed information of nodes and its contents
	Access Table	Concrete access rights for each object
	Users Assignment	Specification of roles (users/credentials)
Definition of the presentation features	Presentation Specifications	Presentation properties for nodes and contents
EVALUATION		
Activity	**Models**	**Concern**
Prototype development	Prototype	Interface mock-up
Evaluation	Evaluation Document	Report about the evaluation process
	Conclusions Report	Improve the system or its design

To ensure completeness, consistency, and integrity among the design models, validation and verification rules are enforced. The ADM models deal with the six design views (navigation, structure, presentation, behavior, processes, and access) using a high-level notation understandable by a multi-disciplinary team. These models are abstractions based on Labyrinth (Díaz, Aedo, & Panetsos, 1997; Díaz, Aedo, & Panetsos, 2001), a hypermedia and platform-independent reference model that makes possible representing the real world and translating this representation into a physical system implemented and distributed using different platforms. The AriadneTool toolkit (Montero, Díaz, Dodero, & Aedo, 2004) automates tasks like creating the ADM models, checking their correctness and integrity, generating documentation about the project development, and creating prototypes in different languages, including

HTML, XML, SMIL, and RDF. The MARAH model is assumed for access modeling.

Basic Features of the MARAH Access Control Model

From a functional point of view, the basic goal of the MARAH model (Aedo et al., 2003) is to provide designers with a series of mechanisms to specify the rules that ensure a proper operation of any hypermedia Web system. Hence, one of its basic assumptions is to use abstractions and concepts that belong to the hypermedia domain in a broad sense so that the model could be integrated into a hypermedia design method such as ADM. MARAH follows an RBAC philosophy in order to simplify management and avoid inconsistencies and errors. Some classical security principles have been assumed in MARAH, including: (1) the system is only accessed through well-formed transactions that are the operations of the Labyrinth hypermedia model; (2) only authenticated users can perform operations, although the authentication process is not considered part of the model but a responsibility of the operation platform (e.g., the Web server); (3) users and processes are assigned the least privilege required to accomplish their objectives; (4) authority is delegated so that not too critical actions are under the author's responsibility; (5) positive and negative authorizations are considered; (6) data abstraction is supported so that security rules are specified in terms of abstractions belonging to the hypermedia domain (such as nodes or contents); and (7) separation of duties is supported. From a structural point of view, the MARAH model is composed by a number of elements and relations among elements that are detailed in Table 2. First, there are a number of **subjects** (S), abstract entities used to represent all those who can access the hypermedia application exercising different permissions. Two kinds of subjects are considered: **roles** (R) and **teams** (T), both of which have a different meaning as well as rather distinct semantics. A role is an organizational position or job function that appears in the application domain, as defined in RBAC models (ANSI, 2004), whereas a team is an aggregation of roles defined to represent existing groups of interest (e.g., Friends of the British Museum), collaborative teams (e.g., development team), or just to alleviate administrative tasks (e.g., system users). Roles and teams support composition mechanisms in order to deal with complex user structures. To gather the complexities of most organizations, hierarchies of roles can be defined as a DAG (Directed Acyclic Graph) where junior roles are specialized into more specific ones using an *is-a* relationship (generalization) involving

Table 2. Components of the MARAH model

Abstract Element	Specification
Subject Roles Teams	$S = \{R \cup T \mid R \cap T = \varnothing\}$ $R = \{r_i,\ i=1..n,\ n \in \mathbf{N}\}$ $T = \{t_i,\ i=1..m,\ m \in \mathbf{N}\}$
Users	$U = \{u_i,\ i=1..p,\ p \in \mathbf{N}\}$
Users assignment	$A = \{<u,r> \mid u \in U,\ r \in R\}$
Separation of duties	$SD = \{<r_i, r_j> r_i, r_j \in R,\ r_i \neq r_j \mid$ $\forall u_k \in U,\ <u_k, r_i> \in A \Rightarrow <u_k, r_j> \notin A\}$
Objects Nodes Contents	$O = \{N \cup C\} \mid N \cap C = \varnothing$ $N = \{n_i,\ i=1..q,\ q \in \mathbf{N}\}$ $C = \{c_i,\ i=1..r,\ r \in \mathbf{N}\}$
Access categories	$SC = \{sc_i,\ i=1..3 \mid sc_i \subseteq sc_{i+1}\ i=1..2\}$
Operations	$Op = \{op_i \mid i=1..s,\ s \in \mathbf{N}\}$
Classification of operations	$\omega: Op \rightarrow SC$
Classification of objects	$\delta: O \rightarrow SC$
Confidentiality	$\psi: O \rightarrow S^n$
Clearance	$\phi: O \times S \rightarrow SC$
Authorization rules	$\forall s \in S, \forall o \in O, \Pi(s,o) = 0$ if $s \in \psi(o)$ $\phi(o,s)$ if $s \notin \psi(o)$
Transition function	$\theta: Op \times O^n \times S \rightarrow O^m$, $(n, m \in \mathbf{N})$ $\theta(op, (o_1, o_2, \ldots, o_N), s)$ is performed iff: a) $\omega(op) \leq \delta(o_i)$, and b) $\Pi(s, o_i) \neq 0$, and c) $\omega(op) \leq \Pi(s, o_i)\ \forall o_i, i=1,\ldots,n$

inheritance. Teams are aggregations of roles and teams, a *whole-part* relationship that allows reference to a number of subjects as a whole. **Users** (U) are assigned one or more roles to be able to use the system through an **Assignment** function (A). A basic and static mechanism of **Separation of duties** is supported by means of a set of pairs of mutually exclusive roles. The **objects** (O) are the **nodes** (N) and **contents** (C) making up the hypermedia application. Nodes are abstract containers of information items, called contents. For example, a Web page is a node containing a number of items such as texts, images, videos, and so on. By separating nodes from contents, fine-grained policies can be supported so that different subjects can have different views of the same nodes. The other basic component of a hypermedia system, the link, is defined as a set of sources and targets called anchors (Díaz et al., 1997), both of which are collections of nodes and/or contents so that they inherit the same rules that apply to their components. Thus, in order to see a link, a subject must be allowed to see all of its anchors. Otherwise, the link will not appear in any node accessed by such user. Two composition mechanisms are supported for nodes and contents: generalization, implying inheritance, and aggregation, used to refer to a group of objects as a whole. The object set takes the form of a DAG by means of two partial orders defined by irreflexive, transitive, and anti-symmetric aggregation and generalization relationships. On the basis of the composition mechanism, the concept of **domain** is introduced to refer to the hierarchical structure defined from an object:

$$\text{domain}(o) = \{o\} \cup \text{domain}(o') \cup \text{domain}(o'') \forall o', o'' \mid o' \in \text{aggregatedBy}(o) \text{ and } o'' \in \text{generalizedBy}(o)$$

$$\forall\, o \in O, \text{aggregatedBy}(o) = \text{list of objects aggregated by } o,$$

$$\forall\, o \in O, \text{generalizedBy}(o) = \text{list of objects generalized by } o$$

Thus, domains are used to refer to a number of objects, irrespective of how and where they are held (as attributes of a relational table, as files, or even as smaller components like tags). The domain concept is more powerful than the physical concept of directory used in some access models. The domain of the root node of a hypermedia system includes all the nodes in the system. A number of generic hypermedia **Operations** (Op) are supported for the different components of the hypermedia Web system such as *createNode*, *createContent*, *placeContentNode*, *createLink*, *activateLink*, and so on. For a comprehensive list of operations, see Díaz et al. (1997). **Access categories** (Sc) are the kinds of access categories supported. A hypermedia Web system is composed of a Basic Hyperdocument, accessed by all the users respecting certain access rules, and a number of Personal Hyperdocuments that are private spaces only accessed and managed by their owners who can be individual users or groups represented by MARAH subject. To deal with these two concepts, three values of access categories, making up a partial order relationship (each one adds permissions to the previous), are considered: (1) Browsing, to retrieve information (nodes and contents) whether selecting links or using other means (maps, search engines, etc.); (2) Personalizing adds the ability to include personal elements (such as private contents, nodes, or links), that is, the possibility of managing personalized spaces; and (3) Editing adds the ability to modify the hypermedia Web system. These categories are used to assign each operation in Op the privileges needed to perform its tasks by means of the **classification of operations** (ω) function. For example, the *createNode* operation requires an Editing category, while *activateLink* is assigned a Browsing one. This classification is hard coded into the specification of the

Labyrinth operations. Objects are assigned an access category to disable any authorization rule temporarily by using the **classification of objects** (δ). Thus, if during the update of a Web site, the security manager assigns a Browsing category to the hypermedia system, authorization rules of subjects will not be considered, and no user will be able to modify any element of the system, irrespective of the permissions the subject is granted. Thus, since no authorizations have to be modified, no erroneous access rights can be exercised. Objects can be assigned negative ACLs where the subjects who are not granted any kind of access are held by means of the **confidentiality** relation (ψ). Positive authorizations are managed by means of the **clearance function** (ϕ) where a subject is granted an access category for an object. Here, the term clearance is not used in its classical meaning (security level assigned to a subject) but as a security level assigned to a subject for a specific object or domain to support context-dependent authorizations. The **authorization rule** (Π) gathers both relationships so provides the manipulation abilities each subject will have in each domain. As MARAH is a model that can be used in different domains of application (e-learning, e-government, etc.), authorizations are defined using objects and generic operations so that MARAH authorizations match the concept of permission in ANSI (2004). Domain-dependent operations (e.g., create course, solve exercise, or access statistics in e-learning) are introduced at a higher level of abstraction (see the Functions in ADM), while MARAH offers generic operations (like *createNode*) applied in different domains (create a course, create a product, etc.). Authorizations are propagated across the subject structure according to the following rules:

R1. Direct propagation of authorization from junior roles

$$\forall s_1 \in S, \forall o \in O \mid s_1 \in \text{generalizedBy}(s_2), s_2 \in S \Rightarrow \Pi(s_1, o) = \Pi(s_2, o)$$

R2. Authorization overriding in senior roles

$$\forall s_1 \in S, \forall o \in O \mid s_1 \in \text{generalizedBy}(s_2), s_2 \in S, \Pi(s_1, o)=x \wedge \Pi(s_2, o)=y, x,y \in \{0\} \cup SC \Rightarrow \Pi(s_1, o)=x$$

R3. Authorization propagation in nested generalization relationships

$$\forall s_1 \in S, \forall o \in O \mid s_1 \in generalizedBy(s_2) \wedge s_2 \in generalizedBy(s_3)\ s_2, s_3 \in S \Rightarrow \Pi(s_1, o) = \Pi(s_2, o)$$

R4. Authorization propagation in parallel generalization relationships

$$\forall s \in S, \forall o \in O \mid s \in generalizedBy(s_i),\ s_i \in S, i=1..n \Rightarrow \Pi(s, o) = \Delta_1^{\,n}(\Pi(s_i, o))$$

$$\Delta_i^{\,n}(\Pi(s,o)) = \begin{cases} 0, \text{if } \Pi(s_i,o)=0, \forall i, i=1,\ldots,n \\ \Pi(s_k,o) \mid \Pi(s_k,o) \geq \Pi(s_i,o), \forall i, i=1,\ldots,n \end{cases}$$

R5. Direct assignment of authorization for team members

$$\forall s_1 \in S, \forall o \in O \mid s_1 \in aggregatedBy(s_2),\ s_2 \in S \wedge \neg\exists\ \Pi(s_1, o) \Rightarrow \Pi(s_1, o) = \Pi(s_2, o)$$

Finally, the **transition function** (θ) is responsible for guaranteeing that only safe operations are performed. This approach separates the policy enforcement point from the operation involved in the access control decision, making the transition function independent of the action to be executed. Every Labyrinth operation (Op) calls to the transition function before its execution so that programmers do not need to be aware of implementing specific code to enforce access control. MARAH is not a closed model as several extensions are foreseen, including a more complex SD mechanism, delegation, or temporal roles. However, it offers a powerful mechanism to specify access rules for hypermedia systems as has been shown in projects like ARCE.

Integrating Security Modeling in ADM: The ARCE Example

The ADM provides a number of design models for access modeling tasks, models which are abstractions of the MARAH components with a graphical and easy to understand notation that can be discussed by all participants in the project. To illustrate its applicability, the ARCE development project is used. ARCE is a Latin American project where the 21 countries making up the Latin-American Association of Governmental Organisms of Civil Defence and Protection are cooperating to create a Web-based platform to improve assistance in disaster mitigation situations (Aedo, Díaz, Fernández, & de Castro, 2002). The ARCE Web system is a platform to share updated and reliable information among the members of the association to orchestrate an integrated and efficient response when a disaster happens (http://www.arce.proteccioncivil.org). The Conceptual Design phase in ADM provides designers with models to specify the system features, taking into account the six design views aforementioned from an abstract point of view. Thus, the system is specified in terms of types of entities (e.g., emergency) instead of concrete elements or instances (e.g., Ivan hurricane) that are defined in the Detailed Design, whether in a declarative or procedural way. ADM evaluation focuses on system improvement by means of a continuous assessment procedure that can be applied to design models or prototypes.

Figure 2. Excerpt of the ARCE Structural Diagram

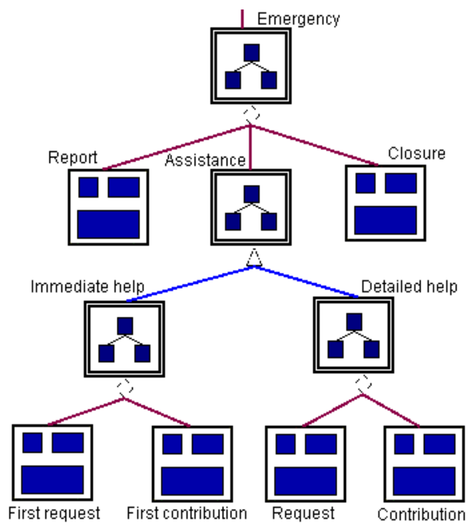
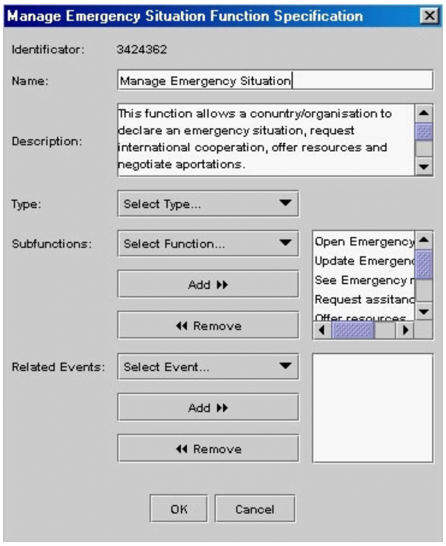


Figure 3. Functional specifications for Manage Emergency ARCE function

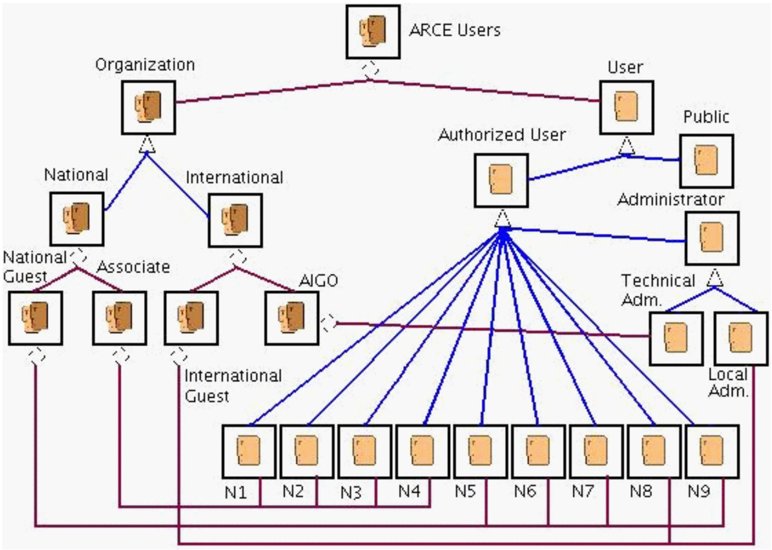


Conceptual Design of ARCE Access Requirements

Definition of the Logical Structure: In this phase, nodes are identified in the *Structural Diagram*. Figure 2 shows part of ARCE Structural Diagram where simple nodes are boxes with a simple border while composites are double border boxes. For instance, the generic concept of emergency in ARCE is composed by three nodes: an initial Report, the Assistance, and the Closure. In turn, Assistance is specialized into the Immediate and the Detailed help (required in the first 48 hours and after, respectively) nodes. Later, nodes will be associated with an access label using the *Categorization Catalogue*, gathering the MARAH classification of objects function. In the Detailed Design, each abstract object is converted into a number of concrete instances.

Study of the System Function: This activity produces the *Navigation Diagram* and the *Functional Specifications*. The former model defines the navigation paths and tools, and the latter defines non-navigation services offered to the users. Functions are operations that belong to the domain of application. Functions can be decomposed into simpler functions. Every Functional Specification includes the function identifier, name, description, type, subfunctions (if any), and related events. For instance, the Negotiate

Figure 4. ARCE users diagram

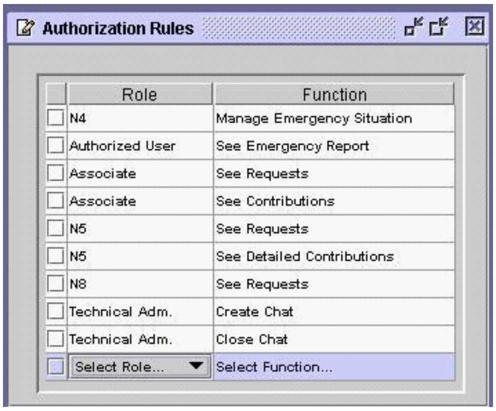


Aportations function, whose specification appears in Figure 3, is divided into Open Emergency, Update Emergency Report, See Emergency Report, Request Assistance and so on. A complete example, including control structures, will be provided in the Detailed Design. In the Specification of Entities activity, the *Internal Diagrams* (spatial-temporal definition of nodes where multimedia compositions are created) and the *Catalogues of Attributes* and *Events* (repositories of properties and behaviors) are created to specify the appearance, navigation capabilities, semantics, and behavior of each node.

Users Modeling: The ADM models subjects using two abstractions: roles and teams, both of which can have attributes increasing their semantics. Hierarchical structures of subjects are represented in the *Users Diagram* model (Figure 4), where the organizational structure of the involved organisms is represented. In the example, the N4 role that represents the operating center of each country is a member of the Associate team, and at the same time, it is a specialization of Authorized user.

Definition of the Security Policy: Finally, in this activity, the access policy is determined by means of the *Categorization Catalogue* and the *Authorization Rules*. The *Categorization Catalogue* assigns to each object the most permissive kind of operation the object will support. Due to the dynamic nature

Figure 5. ARCE authorization rules



Authorization Rules

Role	Function
N4	Manage Emergency Situation
Authorized User	See Emergency Report
Associate	See Requests
Associate	See Contributions
N5	See Requests
N5	See Detailed Contributions
N8	See Requests
Technical Adm.	Create Chat
Technical Adm.	Close Chat
Select Role...	Select Function...

of the ARCE system, the Editing category is assigned to almost every node and content. However, when an emergency is closed, all the objects concerning such a situation are moved automatically to a Browsing category to ensure that no user will modify them. *Authorization Rules* assign each role the functions (s)he can perform. Figure 5 shows the main authorization rules of the ARCE system. As it can be seen, N4 is authorized to manage emergency situations, so (s)he is able to perform all his/her subfunctions, whereas roles like N8 that represent members of invited international organizations (e.g., the UN) can only see the requests. It is important to bear in mind that authorizations are propagated through the subject's structure. Finally, if a subject is not granted a function, the subject cannot initiate it. A subject can be allowed to execute simple or composite functions. As said before, the concept of function is introduced in the ADM, as MARAH does not support domain-specific operations. Each function will give rise to a number of MARAH operations during the Detailed Design when the *Detailed Specification of Functions* are created.

Detailed Design of ARCE Access Requirements

The Detailed Design phase focuses on a lower abstraction level than the Conceptual Design; hence, concrete entities involved in the final system are specified, whether in a declarative or a procedural way. Thus, the first activity is the **Identification of Instances**, which leads to the *Diagram of Nodes*

Instances and the *Instanced Users Diagram*. Each instance will have a special attribute, called the instance descriptor, which makes univocal identification possible. In ARCE, instances of node and user structures are created for each associated country, using the country name as instance descriptor. As described below, the instance descriptor is used to automatically derive concrete authorization rules that apply to instances, instead of to abstract types. For example, there is a node Emergency Situation.Argentina, as well as a role N4.Argentina.

The next activity is the **Specification of Functions**, which produces (1) the *Specification of Access Structures* that contains a detailed description of navigation tools identified in the *Navigation Diagram* of the Conceptual Design phase and (2) the *Detailed Specification of Functions*, where a detailed decomposition of the functions identified in the Conceptual Design is provided. Composite functions are defined in terms of their subfunctions, while simple functions are decomposed into atomic operations involving types of objects. Atomic operations operate on core components that can be nodes, contents, anchors, links, events, users, and attributes (Díaz et al., 1997). Table 3 shows the Detailed Specification of the Create emergency report function. This detailed description provides a link between the *Authorization Rules* of the Conceptual Design and the final *Access Table* of the Detailed Design, as shown in Table 3. The **Specification of Instances** generates (1) a set of *Detailed Internal Diagrams* (where each node and content are fully speci-

Table 3. Detailed functional specification for the simple function "Create emergency report"

```
CreateNode(Report)
#Contents of the report are automatically created
CreateContent(Emergency Type)
CreateContent(Emergency Location)
CreateContent(Damages)
...
#Contents are automatically placed in the report node
placeContentNode(Emergency Type, Report)
placeContentNode(Emergency Location, Report)
...
#User updates of the report contents
1{EditContent(user, Emergency Location)}N
1{EditContent(user, Emergency Type)}N
0{EditContent(user, Damages)}N
...
```

fied—including file references or URIs—so templates can be automatically produced); (2) the *Access Table*; and (3) the *Users Assignment*, the latter of which defines the access policy in terms of concrete subjects and objects. The *Access Table* gathers the access rights each subject instance can exercise with each object instance (that is, concrete nodes and contents of the system). In fact, it translates the entries of the *Authorization Rules* of the Conceptual Design into low-level authorizations defined for objects where access rights are bound to four values: the first three, Browsing, Personalizing, and Editing, to deal with the MARAH clearance function, and NoAccess represents the MARAH confidentiality function. The *Access Table* values are computed automatically from the *Detailed Specifications of Functions* as follows. The subject granted to perform a function is assigned the minimum clearances needed to execute each atomic operation making up the function. For example, if role N4 can execute the function Create emergency report (see Table 3), (s)he requires an Editing category for the node Report, as well as for all the contents included in such node (Emergency Location, Damages, etc.). Since the *Detailed Functional Specification* includes the types of operations required for each object, such low-level permissions can be automatically created in the *Access Table*. But, these access rules are still conceptual insofar as they refer to conceptual nodes and not to concrete instances (e.g., N4 can create an Emergency node). To create concrete access rules, the object and subject instance descriptors are used. Thus, N4.Argentina is granted an Editing right on the Emergency situation.Argentina node while it has a Browsing category for other emergency situations. In this way, context-dependent authorizations can be specified. Finally, the *Users Assignment* assigns concrete users to the roles they can play. If RBAC is adopted as in the ARCE project (a most convenient approach when user authentication is basic requirement in the system), concrete users are assigned specific roles. If credential-based access control were adopted, then the definition of the role would include the set of credential types that must be satisfied by a user to assume such a role. A credential type enforces a set of attributes that users belonging to that type must have, for which regular expressions of the attributes of the roles defined in the *Users Diagram* are used. At runtime, users holding the credentials required to assume a role are automatically assigned to it.

Evaluation of ARCE Access Requirements

In the case of ARCE, design models were evaluated with experts of the association that made possible discussing the access requirements from the beginning of the project and even to detect duplicities and erroneous assignments. Empirical evaluation of prototypes is performed in a regular way proposing an exercise (Aedo et al., 2002), where a country creates an emergency, and the rest of the associates simulate the process of contributing to mitigate its effects. In this way, access rules and the subject structure are iteratively depurated to gather the features and requirements of the different ARCE users.

Conclusions and Future Trends

The integration of access policies into the development process is a key issue to achieve a *holistic* approach in software engineering. In particular, in this chapter we focused on a method called the ADM that unifies different kinds of requirements for hypermedia Web systems, including those related to navigation, presentation, structure, behavior, processes, and access. The ARCE case has shown empirically that integrating access control modeling within the rest of the system models is a powerful and efficient mechanism. From this experience, we learned some useful lessons: (1) Thanks to the integration of access modeling, developers can incorporate access requirements at early design stages using the same abstractions employed to describe other system aspects. In this way, they can integrate solutions to different kinds of requirements before the programming phase is started, avoiding conflicts and providing efficient and smart design options easy to maintain and reuse. (2) The use of visual and understandable design models (ADM diagrams or tables) facilitates communication among the developers and the stakeholders who can take part in the verification process during the design without requiring technological knowledge.

(3) This active, continuous involvement helps stakeholders to think about how responsibilities are actually assigned in the organization, what can derive the redefinition of organizational rules to make them more rational or error solving in the assignment of responsibilities. (4) The tasks of developers become more efficient since AriadneTool offers automation tools as well as validation

mechanisms that control the correctness and completeness of different design models. Otherwise, the overload to interrelate different design views and models becomes unmanageable.

Nevertheless, there are still many open questions for further research. We would like to stress two challenging areas of ongoing work. First, how to integrate design patterns concerning access modeling should be studied, like those reported in Cheng, Konrad, Campbell, and Wasserman (2003), that document solutions that have been proven useful to recurrent problems within a *holistic* software engineering approach. Design patterns tend to bridge between the specification of requirements and the design solutions as far as each requirement can be associated with one or more design patterns that help to solve/worsen such a requirement (Cleland-Huang & Schmelzer, 2003), and, in turn, each pattern is expressed as one or more parts of a design model so that automatic requirements validation could be supported. Another interesting area of research is the development of software engineering methods for Web services requiring the specification of access policies. Here, we can detect at least two different development strategies: tools to develop simple Web services or tools to create composite ones reusing existing simple/composite Web services. In the first case, methods like the ADM can help to design hypermedia-based Web services, integrating requirements concerning navigation, presentation, structure, behavior, processes, and access, though the automation tool should be extended to support the automatic generation of the WSDL descriptions. In the second case, more complex methods are required to assist designers to locate useful Web services and aggregate them in a proper way to satisfy the system requirements, including tools to define the access rules of composite Web services based on the access rules of their components.

Acknowledgments

This work is part of the ARCE++ project (TSI2004-03394) funded by the Spanish Ministry of Science and Education (MEC) and a cooperation agreement between "Universidad Carlos III de Madrid" and "Dirección General de Protección Civil y Emergencias" (Ministry of the Interior).

References

Aedo, I., Díaz, P., Fernández, C., & de Castro, J. (2002). Supporting efficient multinational disaster response through a Web-based system. *Proceedings of the First International Conference EGOV 2002, LNCS 2456* (pp. 215-222).

Aedo, I., Díaz, P., & Montero, S. (2003). A methodological approach for hypermedia security modelling. *Information and Software Technology, 45*(5), 229-239.

ANSI INCITS 359-2004, American National Standard for Information Technology; Role Based Access Control.

Bertino, E., Pagani, E., Rossi, G.P., & Samarati, P. (2000). Protecting information on the Web. *Communications of the ACM, 43*(11), 189-199.

Bertino, E., & Samarati, P. (1995). Research issues in authorization models for hypertext systems. *Proceedings of the 1995 ACM SIGSAC New Security Paradigms Workshop* (pp. 22-27).

Brose, G., Koch, M., & Löhr, K.P. (2001, November 13). *Integrating security policy design into the software development process* (Tech. Rep. B–01–06). Freie Universität Berlin.

Campbell, B., & Goodman, J.M. (1988). HAM: A general purpose hypertext abstract machine. *Communications of the ACM, 31*(7), 856-861.

Cheng, B.H.C., Konrad, S., Campbell, L.A., & Wasserman, R. (2003, September). Using security patterns to model and analyse security requirements. *Proceedings of the High Assurance Systems Workshop (RHAS03),* Monterey Bay, CA.

Cleland-Huang, J., & Schmelzer, D. (2003). Dynamically tracing non-functional requirements through design pattern invariants. *Proceedings of the Workshop on Traceability in Emerging Forms of Software Engineering, IEEE International Conference on Automated Software Engineering.*

Damiani, E., De Capitani di Vimercati, S., Paraboschi, S., & Samarati, P. (2002). A fine-grained access control system for XML documents. *ACM Transactions on Information and System Security, 5*(2), 169-202.

Devanbu, P.T., & Stubblebine, S. (2000). Software engineering for security: A roadmap. In A. Finkelstein (Ed.), *The future of software engineering*. ACM Press.

Díaz, P., Aedo, I., & Montero, S. (2001). Ariadne, a development method for hypermedia. *Proceedings of the Dexa 2001, LNCS 2113* (pp. 764-774).

Díaz, P., Aedo, I., & Panetsos, F. (1997). Labyrinth, an abstract model for hypermedia applications. *Information Systems, 22*(8), 447-464.

Díaz, P., Aedo, I., & Panetsos, F. (2001). Modelling the dynamic behaviour of hypermedia applications. *IEEE Transactions on Software Engineering, 27*(6), 550-572.

Díaz, P., Montero, S., & Aedo, I. (2004). Modelling hypermedia and Web applications: The Ariadne Development Method. *Information Systems*. (In press.)

Fernández, E.B., Krishnakumar, R.N., Larrondo-Petrie, M.M., & Xu, Y. (1996). High-level security issues in multimedia/hypertext systems. In P. Horster (Ed.), *Communications and multimedia security II* (pp. 13-24). Chapman & Hall.

Ferraiolo, D.F., Barkley, J.F., & Kuhn, D.R. (1999). A role-based access control model and reference implementation within a corporate intranet. *ACM Transactions on Information and Systems Security, 2*(1), 34-64.

Garzotto, F., Paolini, P., & Schwbe, D. (1993). HDM: A model-based approach to hypertext application design. *ACM Transactions on Information Systems, 11*(1), 1-26.

Grance, T., Hash, J., & Stevens, M. (2003). Security considerations in the information system development life cycle. NIST 800-64.

Isakowitz, T., Kamis, A., & Koufaris, M. (1998). *The extended RMM methodology for Web publishing* (Tech. Rep. IS98–18). Center for Research on Information Systems.

Joshi, J., Aref, W., Ghafoor, A., & Spafford, E. (2001). Security models for Web-based applications. *Communications of the ACM, 44*(2), 38-44.

Jürjens, J. (2002). UMLsec: Extending UML for secure systems development. *Proceedings of the UML 2002, LNCS 2460* (pp. 412-425).

Lowe, D., & Webby, R. (1999). Utilisation of process modelling in improving the hypermedia development process. *The New Review of Hypermedia and Multimedia, 5*, 133-150.

Montero, S., Díaz, P., & Aedo, I. (2003). A framework for the analysis and comparison of hypermedia design methods. *Proceedings of the IASTED International Conference on Software Engineering (SE 2003)* (pp. 1053-1058).

Montero, S., Díaz, P., Dodero, J.M., & Aedo, I. (2004). AriadneTool: A design toolkit for hypermedia applications. *Journal of Digital Information, Special Issue on Information Design Models and Processes, 5*(2).

Nanard, J., & Nanard, M. (1995). Hypertext design environments and the hypertext design process. *Communications of the ACM*, 38(8), 49-56.

Samarati, P., Bertino, E., & Jajodia, S. (1996). An authorization model for a distributed hypertext system. *IEEE Transactions on Knowledge and Data Engineering, 8*(4), 555-562.

Schwabe, D., & Rossi, G. (1998). An object oriented approach to Web-based application design. *Theory and Practice of Object Systems, 4*(4), 207-225.

Stotts, P.D., & Furuta, R. (1989). Petri-Net-based hypertext: Document structure with browsing semantics. *ACM Transactions on Office Information Systems, 7*(1), 3-29.

Thuraisingham, B. (1995). Multilevel security for information retrieval systems II. *Information and Management, 28*, 49-61.

Zhang, X., Park, J., & Sandhu, R. (2003). Schema based XML security: RBAC approach. *Proceedings of the 17th IFIP 11.3 Working Conference on Data and Application Security,* Estes Park, CO.

Chapter IX

Policy-Based Management of Web and Information Systems Security: An Emerging Technology

Gregorio Martínez Pérez, University of Murcia, Spain

Félix J. García Clemente, University of Murcia, Spain

Antonio F. Gómez Skarmeta, University of Murcia, Spain

Abstract

Network, service, and application management today faces numerous challenges, ones that older ways of doing things cannot solve. The concept of policy-based management (PBM) addresses some of these problems and offers possible solutions. It provides a system-wide view of the network and its services and applications, and shifts the emphasis of network management and monitoring away from specific devices and interfaces toward users and applications. This chapter describes the technology on the policy-based management paradigm which is considered

relevant for providing a common base for researchers and practitioners who need to understand the current status of this emerging technology and how it can be applied to the Web and information systems security field.

Introduction

One of the main goals of policy-based management (Kosiur, 2001; Strassner, 2003; Verma, 2000) is to enable network, service, and application control and management at a high abstraction layer. The administrator specifies rules that describe domain-wide policies which are independent of the implementation of the particular network node, service, and/or application. It is then the policy management architecture that provides support to transform and distribute the policies to each node and thus enforce a consistent configuration in all the elements involved. This is a prerequisite for achieving end-to-end security services or consistent access control configuration in different Web servers, for example.

The use of policies is an intrinsically layered approach allowing several levels of abstraction. There may be, for example, general policies expressing an abstract business goal, and on the other end, there may be policies that express a more specific device, service, or application dependent rule.

Policy rules are independent of a specific device and implementation, but they define a desired behavior in abstract terms. They are stored and interpreted by the policy framework, which provides a heterogeneous set of components and mechanisms that are able to represent, distribute, and manage policies in an unambiguous, interoperable manner, thus providing a consistent behavior in all affected policy enforcement points (i.e., entities where the policy decisions are actually enforced when the policy rule conditions evaluate to "true").

The main functions of policy management architectures are enforcement, that is, to implement a desired policy state through a set of management commands; monitoring, an ongoing active or passive examination of the network, its services, and applications for checking its status and whether policies are being satisfied; and decision making, that is, to compare the current state of the communication system to a desired state described by a policy (or a set of them) and to decide how the desired state can be achieved or maintained.

In the Web and information systems security field, a policy (i.e., security policy) can be defined as a set of rules and practices describing how an organization manages, protects, and distributes sensitive information at several levels.

These different levels range mainly from application to network layer. In fact, on the one hand, users (and developers) are usually aware of the security requirements of the applications and services they are using (or developing) and are willing to express them to the "system" for it to take them into account. On the other hand, administrators of policy-enabled networks are aware of the security services they expect from the network and need some way to define them. Finally, the network (i.e., information system, in general terms) is composed of a set of entities (e.g., routers, firewalls, application servers, Web clients) with certain functionalities and topologies that may also need to be described with another language.

The research community, industry, and standardization bodies have proposed different secure policy specification languages to try to cover these layers either completely (e.g., security policies at the application level) or just in part (e.g., access control policies), but more research seems to be needed in order to find a common *de facto* standard to be used homogenously or, at least, a way to combine and validate several application- and/or network-level specification languages in order to cover all the security aspects of any information system.

A more homogenous approach has been identified regarding the framework needed to manage all this policy information. In this sense, the IETF/DMTF standardization effort (Distributed Management Task Force, 2004; Internet Engineering Task Force, 2004) is considered as a good starting point, although several research works and real implementations have either proposed alternative views or more details on the different building blocks of each of the components of a policy management architecture, that is, network node and/or application server, policy management tool (PMT), policy decision point (PDP), and policy enforcement point (PEP).

As stated before, security policies can be defined to perform a wide variety of actions, from IPsec/IKE management (example of network security policy) to access control over a Web server (example of application-level policy). To cover this wide range of security policies, this chapter aims to examine the current state of policy engines and policy languages, and how they can be applied to both Web security and information systems security to protect information at several levels. Some future directions of interest for the reader are provided as well.

Background

This section is intended to provide the reader with an overview of the main elements and technologies existing in the policy-based management paradigm from both a theoretical and a practical perspective. It starts by describing the kind of security policies that will be considered in this chapter when analyzing policy languages. Then, it describes the process of policy definition, the elements and people involved in it, the different levels of policies, and the main requirements of a policy language. Next, it describes the requirements of policy management architectures. Finally, this section presents a case study related with Business-to-Consumer (B2C) Web-based e-commerce systems, which will be used throughout this chapter to illustrate different concepts, for example, how different security policy languages express the same high-level policy.

Scope of the Security Policy Languages under Consideration in this Chapter

A security policy language can be related to one or several security services. In this chapter, we have reduced our analysis to those languages which include support for any of these security services.

- **Identification/Authentication:** Users are normally identified with their group or associated role, which is stated in some kind of policy language.
- **Access Control:** In this case, policy rules are defined to specify under what conditions a user or an element can access resources, information, or other elements of the target system.

The Policy Definition Process

There are two main perspectives to define a policy (Westerinen et al., 2001), and each represents a different level of abstraction. If a policy expresses a definitive goal, direction, or way of proceeding, this kind of policy will be referred to as a business policy because its context is at the business level. However, if a policy is defined as a set of specific rules to manage devices, services, and/or applications, and therefore is used at the technological level, it is named according to its scope, for example, QoS policy, routing policy, or access control policy. Business policies have a higher level of abstraction and are usually transformed into one or several policies at the technological level.

In the policy definition process (see Figure 1), a human policy maker initially provides the business level policy using an informal language, for example, "*Customers require strong protection for e-commerce online payments.*" Then, the task of a policy administrator is to transform the business policies into implementable policies using a formal representation. To do so, the administrator uses a policy language that assures the representation of policies will be unambiguous and verifiable. Other important requirements of any policy language are:

- **Clear and well-defined semantics:** The semantics of a policy language can be considered as well-defined if the meaning of a policy written in this language is independent of its particular implementation.
- **Flexibility and extensibility:** A policy language has to be flexible enough to allow new policy information to be expressed and extensible enough to allow new semantics to be added in future versions of this language.
- **Interoperability with other languages:** There are usually several languages that can be used in different domains to express similar policies, and interoperability is a must to allow different services or applications from these different domains to communicate with each other according to the behavior stated in these policies.
- **Support of multiple bindings:** It has to be possible to convey policy instances in a number of different protocols; some of the most relevant for performing management tasks are SNMP (Simple Network Management Protocol) (Levi, Meyer, & Stewart, 2002), COPS (Common Open Policy Service) (Durham et al., 2000), LDAP (Lightweight Directory Access Protocol) (Wahl, Howes, & Kille, 1997), WSDL (Web Service Description Language) (Christensen, Curbera, Meredith, & Weerawarana, 2001), SOAP (Simple Object Access Protocol) (Mitra, 2003), DIAMETER (Calhoun et al., 2003), and HTTP (Hyper Text Transfer Protocol) (Fielding et al., 1999).
- **Amenable to combining:** A policy language should include a way of grouping policies.

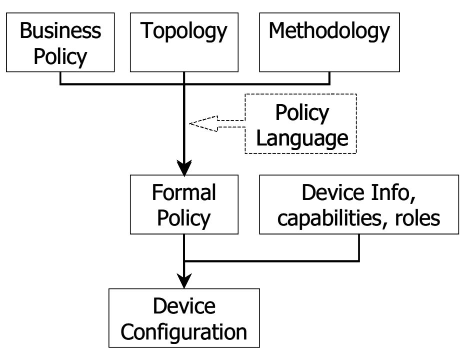
In addition, the translation between the business policy and technological level policy may require additional information such as the network topology and methodology. The topology of the network consists of the set of network devices under management, while the methodology decides the principles that should be used (e.g., decide between the use of public key cryptography or symmetric key cryptography). Finally, the formal policy is translated into device configuration based on the device particular information such as its capabilities and roles.

The most usual policy format defines a policy as a group of policy rules. Each policy rule is the binding of a set of actions to a set of conditions. If the conditions are satisfied, the actions will be enforced. Otherwise, the actions will not be performed.

The administrator has the responsibility of the policy management of an administrative domain. The administrative domain may include various routers, servers, hosts, access points, and switches under the same management authority and/or procedure. Moreover, the policy rules are defined in a domain application and at a functional level, which is called policy scope. For example, we can talk about Security Policies where Security is the policy scope. Similarly, we can talk about QoS Policies, Access Policies, Routing Policies, and so forth.

From a practical perspective, the administrator defines and controls the policies of an administrative domain from a Policy Console. The management of policies will be performed through this console as a user interface (usually a

Figure 1. The policy definition flow



Web browser) to allow definition of policies and monitoring of the status of the policy targets. The policy target is an entity, or set of entities, whose behavior is defined using policies. They are grouped by roles; the main reason for this is that policies are defined not per individual target, but on a per-role basis. This allows us to manage and configure devices as a whole and not as a collection of individual devices. A role may be assigned to different targets, and each target may play several roles.

The Policy Management Process

Once the policy has been defined for a given administrative domain, an architecture is needed to transfer, store, and enforce this policy in that domain. The main requirements for such policy management architecture are:

- **Well-defined model:** Policy architectures need to have a well-defined model independent of the particular implementation in use. In it, the interfaces between the components need to be clear and well-defined.
- **Flexibility and definition of abstractions to manage a wide variety of device types:** The system architecture should be flexible enough to allow addition of new types of devices with minimal updates and recoding of existing management components.
- **Interoperability with other architectures (inter-domain):** The system should be able to interoperate with other architectures that may exist in other administrative domains.
- **Conflict detection:** It has to be able to check that a given policy does not conflict with any other existing policy.
- **Integrating the business and networking worlds:** The system should be able to map business policies (i.e., high-level policies) into middleware/network policies (i.e., low level policies).
- **Scalability:** It should maintain quality performance under an increased system load.

A Case Study

Once we have described the basic concepts of policy management and the requirements of policy languages and architectures, this background analysis will be complemented with one case study, which will be used throughout the chapter to show the advantages and also some of the current open issues of using this approach to management as compared to the traditional way of managing.

Business-to-consumer (B2C) e-commerce systems are an example of Web Information System (WIS) where security is a fundamental aspect to be considered. A secure e-commerce scenario requires transmission and reception of information over the Internet such that:

- It is accessible to the sender and the receiver only (privacy/confidentiality).
- It cannot be changed during transmission (integrity).
- The receiver can be sure it came from the sender (authenticity of the source).
- The sender can be sure the receiver is genuine (authenticity of the destination).
- The sender cannot deny (s)he sent it (non-repudiation).

As stated before, security policies can be specified at different levels of abstraction. The process starts with the definition of a business security policy, for example, "*Customers require strong protection for e-commerce online payments*."

Next, the policy security policy is usually expressed by a policy administrator as a set of IF-THEN policy rules, for example:

IF (Connection between <Customer-Computer> and <Payment-Server>)
THEN (Apply High-Level Security)

Subsequently, the administrator assigns this policy to a role (or a combination of roles); in this case, the role is "Payment Server". The policy system validates the formal policy, stores it, and applies it automatically to each device with this role. It is then in charge of returning monitoring information to the administrator, who can decide if the policy has been well applied to the payment servers and has the desired effect.

At the middleware and network levels (i.e., low level), the security specification enables us to represent and verify the constraints in the specific payment servers. Low-level policy representations are generally procedural and specify the logic of how to achieve the goal. The policy languages we will be analyzing in the discussion section are able to specify security at middleware and network levels and will be used to provide policy examples related to this case study.

Discussion

Now that the basic concepts of this paradigm have been presented and a case study has been introduced, this section shows how the administrator models the business security policies using a Security Policy Representation (Damianou, Bandara, Sloman, & Lupu, 2002) to create a formal representation of the business policy and examples of architectures allowing the management (definition, enforcing, and monitoring) of such policies.

Although many security policy specifications exist, we have selected three of them (Ponder, XACML, and PCIM/CIM) as they are currently considered promising alternatives and fit well with the policy language requirements stated above. Indeed, they have most of the components required in one "ideal" security policy language.

Security Policy Languages

Description of Selected Languages: Ponder, XACML, and PCIM-Based Languages

Ponder (Damianou et al., 2001) is a declarative, object-oriented language developed for specifying management and security policies. Ponder includes constructs for specifying both primitive and composite policies. Primitive policies are able to express authorizations, obligations (stating that a user must or must not execute or be allowed to execute an action on a target element),

Figure 2. Example of policy expressed in Ponder

```
inst oblig SecureOnlinePayment {
        on        OnlinePaymentUse(clientid, merchantid)
        subjec    s = EndUser
        target    t = Traffic
        do        securecommunication (clientid, merchantid, strong) | error()
}
```

information filtering (used to transform input or output parameters in an interaction), refrain policies (which define actions that subjects must not perform on target objects), and delegation policies (which define what authorizations can be delegated to whom).

Ponder composite policies facilitate policy management in large and complex administrative domains. They are used to define a syntactic scope for specifying a set of related policies, or they simply provide reusability of common definitions. There are three types of composed policies: roles, relationships, and management structures.

Figure 2 presents an example of policy expressed in Ponder using our case study.

It shows how to express with an obligation policy the business level security policy and IF-THEN rules defined in our case of study. The obligation policy reacts to an event (following the on keyword) that enables us to detect the execution of an online payment from the client identified by its clientid to the merchant identified by its merchantid. The action follows the do keyword. Two actions are specified and separated by |. This operator indicates that the second action (error()) is performed only if the first action fails. The purpose of the first action is to provide a strong protection for the communication between the client and the merchant.

Ponder is also able to deal with meta-policies, which specify a permitted value for a valid policy. Although it currently seems to be one of the most complete and fully tested representations of a policy management system, it lacks interoperability with standard proposals with similar objectives (management and authorization policies) such as CIM, PCIM, or XACML. Though, some interesting work is currently being undertaken by the Imperial College and CISCO in the Polyander Phase II project (Polyander: Phase 2, 2004).

The second security policy language under analysis, eXtensible Access Control Markup Language (XACML) (OASIS, 2003), is an XML specification from OASIS (Organisation for the Advancement of Structured Information Standards) for expressing policies for information access over the Internet.

XACML is used to specify a subject-target-action-condition access control policy in the context of a particular XML document. The notion of subject comprises identity, group, and role, and the granularity of target objects can be as fine as individual document elements. The language supports roles defined as a collection of attributes relevant to a principal.

XACML describes both an access control policy language and a request/response language. The policy language provides a common means to express access control policies, and the request/response language expresses queries about whether a particular access should be allowed (requests) and describes answers to those queries (responses).

In a typical XACML usage scenario, a subject (i.e., user or software application) wants to take some action on a particular target (i.e., resource). The subject submits its query to the entity managing the resource, which is called policy enforcement point (PEP). It forms an XACML request message based on the attributes of the subject, action, target, and any other relevant information and sends it to a policy decision point (PDP). The PDP analyzes the request and determines whether access should be granted or denied according to the XACML policies that are applicable to this request. Then, an XACML response message is returned to the PEP, which will then allow or deny access to the subject.

The specific policy language class model used in version 1.1 (the current XACML standard version, although 2.0 is already a working draft) can be accessed in OASIS (2003). Its main components are rules, policy, and policy-set.

The main advantage of XACML is that it uses XML. This technology is a widely adopted standard with browsers that can be used to display and edit the policy specification. XACML also provides for fine-grained control of activities (such as read, write, copy, delete) based on several criteria, including the following: attributes of the user requesting access (e.g., "Only administrators can view this document"), the protocol over which the request is made (e.g., "This data can be viewed only if it is accessed over HTTPS"), and the authentication mechanism (e.g., "The requester must have been authenticated using a digital device").

From our point of view, the main failing of XACML is that the policy is rather verbose and not really aimed at human interpretation. In fact, these are the two main reasons for not including a complete example here. In addition, the language model does not include a way of grouping policies.

The last approach under analysis is policy core information model (PCIM) (Moore, Ellesson, Strassner, & Westerinen, 2001). PCIM and its extension (PCIMe) is a notable work by the Internet Engineering Task Force (IETF) to define an object-oriented information model for representing policy information. The classes comprising PCIM are intended to serve as an extensible class hierarchy (through specialization) for defining policy objects that enable policy administrators to represent policies of different types (IPsec, QoS, etc.). The Policy Model enables construction of policy rules of the form:

if <condition(s)> then <action(s)>

The <condition(s)> term is a Boolean expression used to specify the rule selection criteria. These criteria may include temporal, scoping, and state-related conditions. If the set of conditions associated with a policy rule evaluates to TRUE, then the set of actions is executed. Although this type of policy rule has similar semantics to an obligation policy with the form event-condition-action, there is no explicit event specification to trigger the execution of the actions.

PCIM and PCIMe extend the common information model (CIM) version 2.7 (Distributed Management Task Force, 2004) defined by the DMTF with classes to represent policy information. The CIM provides abstractions and representations of the entities involved in a managed environment including their properties, operation, and relationships.

Figure 3 presents an example of the type of rule that administrators can specify using PCIM. It is related with the case study described earlier.

This low level policy is the result of a mapping of the high-level policy in a particular environment where IPsec is used to provide security at the network layer. The policy described in this example requires the use of the IPsec ESP protocol and proposes several algorithms (3DES for confidentiality and SHA-1 or MD5 for integrity), the tunnel mode and the anti-replay protection to secure the communication between the IP addresses 2.2.2.2 and 3.3.3.3. The particular payment application is identified in this example using a port number (8000).

Figure 3. Example of policy expressed in PCIM

IF SourceIPAddress = 2.2.2.2 and SourcePortNo = 8000 and
DestinationIPAddress = 3.3.3.3 and DestinationPortNo = 8000
THEN CONNECT with bi-directional and unicast among 2.2.2.2 at 8000 and
3.3.3.3 at 8000 with ESP with (3DES) and (SHA-1, MD5) and tunnel and
Anti-Replay

Comparative Analysis of the Security Policy Languages Described

Table 1 shows a comparison of the aforementioned security policy languages. Many aspects can be identified as part of this comparison, although the most relevant are:

- The list of security services supported.
- **Policy release:** Another important dimension for the security specification is the policy release. This permits us to identify how policies are triggered when expressed using a given policy language. In this sense, languages can be classified as proactive or reactive. In proactive languages policies are represented by the condition(s) that need to be verified before executing the action(s). In reactive languages, policies are triggered when certain system events occur. These are also called event-based languages.
- **Policy representation technique used by the language:** In the case of PCIM, as it is an information model, several techniques can be used by each particular language based on it. The IETF has been making efforts to express these languages as LDAP schemas, while some recent works use PCIM as XML schemas.
- **Types of groups:** Three types of groups have been identified:
  - **Domain:** A domain generally groups elements together in order to apply a common set of policies on them. The main goal for defining these domains is to reduce the number of policies that need to be applied. A set of policies is no longer associated to one element but to a group of elements. Elements can be grouped together according to geographical boundaries, object type, responsibility and authority, or at the convenience of the managers.

Table 1. Comparative analysis between Ponder, XACML, and PCIM-based security languages

Languages/Dimensions	PONDER	XACML	PCIM-based
Security Policy	Access Control Authentication Delegation Obligation Refrain	Access Control	Access Control Authentication
Policy Release	Reactive/Proactive	Reactive	Reactive
Representation	Functional blocks	XML	LDAP/XML/Others
Types of groups	Domains Roles Group	Domains Roles	Domains Roles

- **Role:** A role is linked to a set of rules describing the rights and duties associated with it. It usually refers to a position in a system.
- **Group:** The Ponder language describes another way of grouping policies together: the group. A group in Ponder represents a set of policies and constraints with some semantics relationship (e.g., relating to a particular service or application).

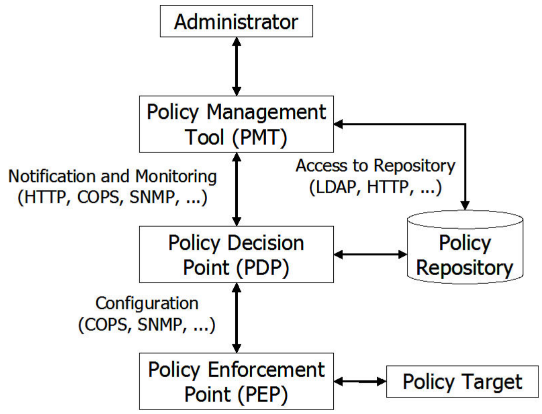
Architecture and Functions

This subsection presents different architectural approaches allowing automatic and distributed management (definition, enforcing, and monitoring) of security policies. The approach of the IETF has been traditionally considered as a basic architecture. In fact, the research community and the standardization bodies have defined some other approaches including new elements to deal efficiently with specific security services. Two of the most relevant proposals are presented here.

The Approach of the IETF

The basic approach for a policy-based management architecture was proposed by the Working Groups of the Internet Engineering Task Force (IETF).

Figure 4. The IETF policy management architecture



The proposed system (Yavatkar, Pendarakis, & Guerin, 2000) by the RAP Working Group of the IETF to manage networks based on policies was first developed in the context of integrated services, where a policy framework manages the admission control. This proposal is independent of any service model, so it was soon thought that it could also be used in QoS with Differentiated Services (Chan, Sahita, Hahn, & McCloghrie, 2003) or security (Blaze, Keromytis, Richardson, & Sanchez, 2003).

Figure 4 shows the four main functional elements of the IETF policy framework. They are also used as part of most of the policy management architectures identified as part of this research, as is the case of the XACML and UPM (Unified Policy Management) policy architectures which are outlined below.

- The Policy Management Tool (PMT) allows the administrator to create, edit, and delete policies and monitor the status of the policy-managed environment.
- The Policy Repository is a storage that is used by the management tool (PMT) to store the policies and by the decisions points (PDPs) to recuperate them. The IETF suggests the use of a Lightweight Directory Access Protocol (LDAP), but other possible solutions such as other directories or databases are also available.
- The Policy Decision Point (PDP) has a view over the whole network area under control, and it is responsible for interpreting the policies stored in

the repository, recuperating the set of rules needed by each PEP, transforming them into a format that can be understood by the PEP and distributing them to the PEP. Usually the PDP is responsible for a specific set of policies, for example, Web Access Control policies.

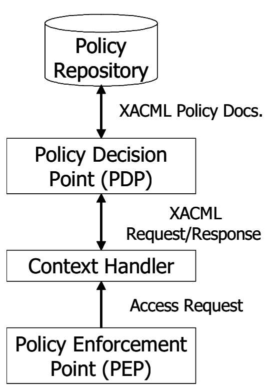
- The Policy Enforcement Point (PEP) is a component running on a network node that can apply and execute the different policies. PEPs can also inform the PDP when any unknown situation is produced.

The IETF framework establishes COPS (Durham et al., 2000) to transfer policy decisions to the PEPs and to transfer requests and reports from PEP to the PDP. The model is open to other mechanisms such as HTTP/HTTPS, FTP, or SNMP (Levi et al., 2002). Furthermore, data in the Policy Repository will change from time to time, and PDPs need to be informed of these changes as they happen. This framework does not specify the means for notifying changes. There are a number of possible ways to make this notification (e.g., polling, LDAP change notification, SNMP traps, etc.).

A Policy Architecture for Access Control

The XACML policy management architecture shown in Figure 5 is a clear example of extension of the IETF model for a particular security service: access control.

Figure 5. The XACML access control policy management architecture



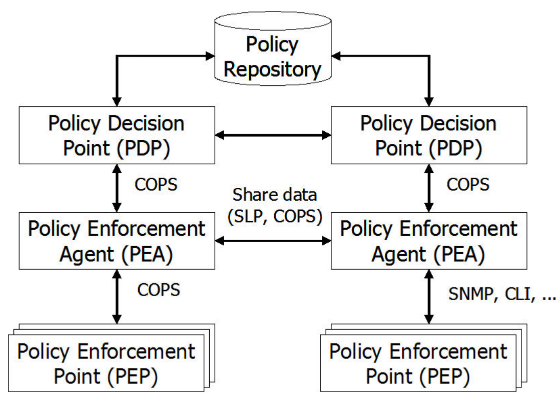
In this architecture, the PDP answers authorization queries based on the resource policies. The *context handler* is implemented as an authorization module for the PEP. It generates authorization requests based on the resource requests that the PEP receives from the network node and provides for communication between the PDP and the PEP.

UPM (Unified Policy Management)

The Unified Policy Management (UPM) (Eddie-Law & Saxena, 2003) proposal also represents an extension of the IETF architecture for policy management. UPM is a multi-tier policy management architecture with certain similarities to a management system based on agents. In a policy domain, there are (apart from the policy repository) three tiers in this architecture (as shown in Figure 6). The top tier is where the policy decision points (or policy servers) are located, and these perform the policy decisions. The middle-tier layer coordinates the communications and creates transparency between network nodes and the policy decision points. The entity located in this middle tier is known as the Policy Enforcement Agent (PEA). If the network nodes do not understand the policy decisions, the PEA is responsible for policy rules translation and executions.

The policy server obtains a service request through either the COPS request from the network node or from the policy administrators with domain level decision. Policy rules are obtained from the policy repository, and the PDP will

Figure 6. The UPM policy management architecture



finalize a to-be-applied policy rule and deliver this decision to the PEA at the middle tier. Each PEA at the middle tier enforces all policy rules from the PDPs. The primary functions of the PEA include: (1) relaying protocol messages between PDP and PEPs; (2) translating different policy rules; (3) distributing different policy sessions to PDPs; (4) ensuring that policy is applied; and (5) informing PEPs about other PEAs.

One enhancement in UPM is that it provides load balancing and load sharing mechanisms at the middle and upper tiers. When there are numerous network nodes to control, there may be multiple PEAs in a single policy domain. The system provides inter-PDP communications, inter-PEA communications, and PDP-PEA communications.

This UPM design permits a concept of unified information model (UIM) between the policy servers and the middle-tier entities. This design unifies the system design at both the policy server and the middle-tier entity. Therefore, we can add more PDPs and PEAs at any time, if required, without disrupting any existing policy controlled services.

There are some important design achievements in the UPM model:

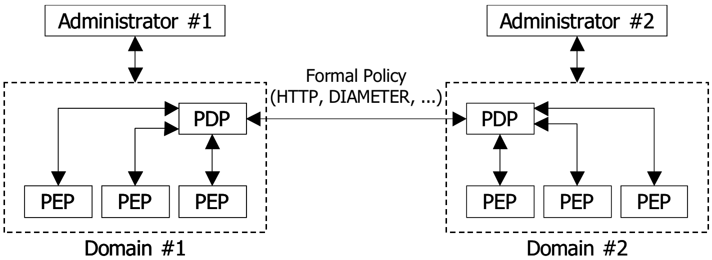
- The load balancing and sharing mechanisms enhance system scalability.
- The system reduces the points of failure at both the middle and upper tiers, hence, enhances reliability.
- All communication above the middle tier is via a common protocol (COPS) using common message formats.

Policy Architectures for Multi-Domain Scenarios

So far, the discussion of this section has focused on only one administrative domain. When there are several administrative domains, and they want to exchange policy information, new problems appear such as the policy conflict detection between domains and the policy representation and the transport protocol to be used.

If all administrative domains use the same policy representation, the exchange of policy information is seamless and direct. However, if the representation is different, it is necessary to agree on a common representation. Following the discussion in the previous section, the policy representation based on the IETF/DMTF information model is the most appropriate for this task. On the other

Figure 7. Exchange of policies between different administrative domains



hand, the model is open to several transport protocols such as HTTP, Diameter, or SNMP (as shown in Figure 7), although it seems that Diameter (Calhoun et al., 2003) is gaining acceptance in the research community on inter-domain scenarios.

There are a number of difficulties with the IETF-based policy management architectures presented above. In particular, they are usually quite generic and broad, and there is no specific policy specification language that is standardized (except in the case of XACML for access control policies), so the interoperability of policy systems and conflict detection is not defined. However, these architectures (the IETF-based ones) can be considered advantageous for a number of reasons:

- They reduce the complexity and scale of the management problem and automate the management tasks (Verma, 2002).
- They introduce the ability of the management system to adapt its behavior dynamically to different environmental conditions.
- They allow the system to achieve higher level objectives such as business goals and service-level agreements (SLAs).
- They improve the level of security through the use of authorization policies and explicit specification of the behavior of security components within the network through obligation policies (Sloman & Lupu, 2002).

Conclusions and Future Trends

This chapter has provided the current trends of policy-based management applied to the protection of information over the Web at several levels (network, information systems, Web servers, etc.). It has described some initial concepts related with policy management complemented by a case study where the reader can see the advantages and also some of the current open issues of using this approach to management vs. traditional approaches.

This chapter has also provided some discussions of the most relevant security policy specification languages and information models. Our perspective on the main issues and problems of each of them has also been presented, based on different criteria such as their ability to express security aspects at the application or the network level (or both), the representation technique they use, or the concepts they are able to express.

According to our analysis, XACML, Ponder, and PCIM/PCIMe are the predominant representations in specifying management and security policies. Ponder and XACML are policy languages, whereas PCIM is an information model allowing different policy semantics to be expressed. PCIM/PCIMe have been incorporated in the latest versions of the Common Information Model (CIM) defined by the DMTF, including improvements to the class hierarchy.

CIM Policy Model emphasizes the definition of general event-condition-action semantics. These semantics are conveyed in an abstract fashion, independent of any policy language or implementation. This approach will allow the mapping of the CIM Policy Model onto an explicit policy language such as PONDER or XACML, and vice versa.

A discussion on the main architectural components for managing security policy specifications has also been presented. These are motivated from the basic set of functions (i.e., enforcement, monitoring, and decision making) that need to be implemented in an administrative domain to fulfill certain policy-based behavior.

Based on our findings, we see the state-of-the-art as being characterized to use a certain number of well-known languages with different purposes, sometimes applicable at different levels, and not yet interoperable between them, standardized IETF management protocols (e.g., COPS, SNMP, DIAMETER) and certain architectural components (i.e., PDP, PEP, PMT) which allow for multi-vendor solutions but address mostly single-domain networks.

Regarding future trends in policy-based management, our state-of-the art analysis indicates that policy-based management of Web and information systems security is of importance to establish a communication infrastructure which is flexible, dynamic, and has a nice price-performance ratio. However, our analysis also indicates that average policy-based systems are still based on proprietary solutions (although some standard languages and protocols are starting to be used), and they address mostly single-domain networks, which is far from the final objective of having management products addressing multi-vendor, multi-service scenarios with open standardized protocols and interfaces.

Thus, important research and development work is still needed in order to identify the most promising solutions to a certain number of open technical issues. Research on policy conflict resolution, interoperability (in multi-domain scenarios), and automated policy translation (from high-level abstract policies to low-level executable policies and vice versa) seem to be steps to be taken in this direction.

Acknowledgments

The authors would like to thank to the reviewers and editors for many valuable comments and suggestions. This work has been partially funded by the EU POSITIF (Policy-based Security Tools and Framework) IST project (IST-2002-002314).

References

Blaze, M., Keromytis, A., Richardson, M., & Sanchez, L. (2003, August). IP security policy (IPSP) requirements. Request for Comments 3586. *IETF Network Working Group*.

Calhoun, P., Loughney, J., Guttman, E., Zorn, G., & Arkko, J. (2003, September). Diameter Base Protocol. Request For Comments 3588. *IETF Network Working Group*.

Chan, K., Sahita, R., Hahn, S., & McCloghrie, K. (2003, March). Differentiated services quality of service policy information base. Request For Comments 3317. *IETF Network Working Group*.

Christensen, E., Curbera, F., Meredith, G., & Weerawarana, S. (2001, March). Web services description language (WSDL) 1.1. W3C Note. *World Wide Web Consortium*.

Damianou, N., Bandara, A.K., Sloman, M., & Lupu, E. (2002). *A survey of policy specification approaches.* Imperial College, Department of Computing, London.

Damianou, N., Dulay, N., et al. (2001). The Ponder Policy specification language. *Proceedings of the Policy 2001: Workshop on Policies for Distributed Systems and Networks*.

Distributed Management Task Force, Inc. (2004). DMTF standards and initiatives.

Durham, D., Boyle, J., Cohen, R., et al. (2000, January). The COPS (Common Open Policy Service) protocol. Request For Comments 2748. *IETF Network Working Group.*

Eddie-Law, K.L., & Saxena, A. (2003). Scalable design of a policy-based management system and its performance. *IEEE Communications Magazine, 41,* 72-79.

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., & Berners-Lee, T. (1999, June). Hypertext Transfer Protocol—HTTP/1.1. Request For Comments 2616. *IETF Network Working Group*.

Internet Engineering Task Force. (2004). Policy Framework Working Group.

Kosiur, D. (2001). *Understanding policy-based networking.* John Wiley & Sons.

Levi, D., Meyer, P., & Stewart, B. (2002, December). Simple network management protocol (SNMP) applications. Request For Comments 3413. *IETF Network Working Group.*

Mitra, N. (2003, June). SOAP Version 1.2 Part 0: Primer. W3C Recommendation. *World Wide Web Consortium*.

Moore, B., Ellesson, E., Strassner, J., & Westerinen, A. (2001, February). Policy Core Information Model. Request For Comments 3060. *IETF Network Working Group.*

OASIS. (2003, August). Extensible Access Control Markup Language (XACML), version 1.1.

Polyander: Phase 2. (2004). Imperial College, Department of Computing.

Sloman, M., & Lupu, E. (2002). Security and management policy specification. *IEEE Network*.

Strassner, J. (2003). *Policy-based network management: Solutions for the next generation*. Morgan Kaufmann.

Verma, D.C. (2000). *Policy-based networking: Architecture and algorithms.* Pearson Education.

Verma, D.C. (2002). Simplifying network administration using policy-based management. *IEEE Network, 16,* 20-6.

Wahl, M., Howes, T., & Kille, S. (1997, December). Lightweight directory access protocol (v3). Request For Comments 2251. *IETF Network Working Group*.

Westerinen, A., Schnizlein, J., Strassner, J., et al. (2001, November). Terminology for policy-based management. Request For Comments 3198. *IETF Network Working Group.*

Yavatkar, R., Pendarakis, D., & Guerin, R. (2000, January). A framework for policy-based admission control. Request For Comments 2753. *IETF Network Working Group*.

Chapter X

Chinese Wall Security Policy Model: Granular Computing on DAC Model

Tsau Young Lin, San Jose State University, USA

Abstract

In 1989, Brewer and Nash (BN) proposed the Chinese Wall Security Policy (CWSP). Intuitively speaking, they want to build a family of impenetrable walls, called Chinese walls, among the datasets of competing companies so that no datasets that are in conflict can be stored in the same side of Chinese walls. Technically, the idea is: $(X, Y) \notin CIR$ (= the binary relation of conflict of interests) if and only if $(X, Y) \notin CIF$ (= the binary relation of information flows). Unfortunately, BN's original proof has a major flaw (Lin, 1989). In this chapter, we have established and generalized the idea using an emerging technology, granular computing.

Introduction

Recent events, such as e-commerce and homeland security, have prompted us to revisit the idea of the Chinese Wall Security Policy Model (Lin, 2001). "The Chinese wall policy combines commercial discretion with legally enforceable mandatory controls...perhaps, as significant to the financial world as Bell-LaPadula's policies are to the military" (Bell, 1987, p. 000). This is asserted in the abstract of Brewer and Nash's (BN's) (1989) article. It is still valid today.

Background

Chinese Wall Security Policy (CWSP) Model

Let us start with recalling the proposal of Brewer and Nash (BN). In 1989, BN proposed a very intriguing commercial security model, called Chinese Wall Security Policy (CWSP) Model. Intuitively speaking, the idea was to build a family of impenetrable walls, called Chinese walls, among the datasets of competing companies so that no datasets that are in conflict can be stored in the same side of Chinese walls. The intent of the proposal was a good one. In their model, BN assumed the set O of corporate datasets could be partitioned into pairwise disjoint subsets, called conflict of interest (CIR) classes. Such a collection of pairwise disjoint subsets is referred to in mathematics as a partition and is known to induce an equivalence relation and vice versa (see for example, Brualdi, 1992). So, BN has assumed CIR is an equivalence relation that is a reflexive, symmetric, and transitive binary relation. Considering the real-world meaning, would *conflict* be reflexive? Appealing to common sense, there is no dataset that is self-conflict, so CIR is unlikely an equivalence relation. Observing this fact, in the same year, we presented a modified model at the Aerospace Computer Security Application Conference; the model was called Aggressive Chinese Wall Security Policy (ACWSP) model (Lin, 1989b). In that paper, we did not bring out the essential strength of the ACWSP model. A relatively inactive decade has passed. Due to the recent development of granular computing, we refined the idea of ACWSP and successfully captured the *intuitive intention* of BN theory and outlined it in COMPSAC and RSCTC (Lin, 2002a, b). *Though the collection of CIR-classes is not a partition, it* *is a binary granulation*. Roughly, a binary granulation of U is a collection of subsets, called granules, that is not quite a partition but the set of the center (or core) of each granule forms a partition; see the main text below. In terms of binary granulation, we can capture the spirit of Chinese wall security policy (CWSP). The methodology is more than CWSP; it has profound impacts on the analysis of information flow in DAC.

With mild assumptions on DAC (Discretionary Access Control) model, we can actually regulate *malicious* Trojan horses so that information, no matter how it flows, will not flow into "enemy" hands; it will flow only among "friends".

Granular Computing

Mathematically, DAC model has a structure, that is, a subset of binary neighborhood systems (BNS), or binary granulations (Lin, 1988, 1989a, 1998a). Seymour Ginsburg observed that this is a list structure (verbally 1989); however, we have not used the list structure because a list may involve linear ordering. The study of such a structure has gained much attention recently and is named granular computing. It has originated from four facets: Let us speak in the chronological order. The first one is Hsiao. In his attribute based database model, Hsiao clustered the attribute domain into semantically related granules (equivalence classes) (Demurjian & Hsiao, 1988; Hsiao & Harary, 1970; Lin, 1992; Wong & Chiang, 1971). Clustering is an important technique in database theory; it partitions a dataset such that semantically related data are stored in physical proximity. The second one, probably the deepest one, is used in the design of fuzzy logic control systems. It decomposes the input space into finite fuzzy granules (Lin, 1996, 1997). The explicit discussion of such a notion of granularity, scientifically (for a different purpose), is in the article of Zadeh (1979) and, more recently, in (Zadeh, 1998, 2002). The third groups are from the theory of data. Pawlak (1982) and Lee (1983) nearly simultaneously observed independently that attributes of a relation induce partitions on the set of entities (Pawlak, 1982; Lee, 1983) and studied relational tables from such observations. Pawlak called his idea rough set theory, while Lee called his study algebraic theory of relational databases. The last faucet comes from approximate retrieval (Chu & Chen, 1992; Lin, 1988, 1989a; Motro, 1986). For developing a theory of approximate retrieval in databases, Motro, in essence, introduced the notion of metric space into databases. We observed that the notion of metric spaces does not naturally exist in a data model, so we imported the notion of (pre-) topological spaces into the attribute domains and called them neighborhood systems (Lin, 1988, 1989a, 1998a). This view leads to a geometric study of granulations (Lin, 1998a, b, 1999). The notion of neighborhoods (granulation) can be traced back to Cauchy's epsilon and delta idea. However, the current focus is on the computation side. This methodology that includes rough set theory has far-reaching consequences in various fields, for example, data mining, security, text mining, and bioinformatics (Lin, 2002b, c; Lin & Chiang, 2004, 2005; Liu et al., 2004). In this chapter, we will focus on security. We would like to note that Pawlak (1984) also discussed the notion of conflict; however, his did not consider it as a binary relation but a special form of relational table, called information system.

The Binary Relation and Conflict of Interests (CIR)

First, we will recall some definitions about binary relations:

Definition. Let $B \subset V \times V$ be a binary relation.

1. B is symmetric, if $(u, v) \in B$ implies $(v, u) \in B$.
2. B is anti-reflexive if B is non-empty and $(x, x) \notin B$, for all $x \in V$. That is, $B \cap \Delta = \emptyset$, where D= $\{(x, x) \mid x \in V\}$ is called diagonal set.
3. B is anti-transitive if B is non-empty and if $(u, v) \in B$ implies that $\forall w \in V$ one and only of (u, w) or (w, v) belongs to B.
4. The complement of B, $B' = V \times V \sim B$, is called the complementary binary relation (CBR) of B.

For intuitive meaning of these terms, see the next section.

Let us recall some analysis from Lin (1989b). Let O be the set of corporate data under consideration. In Brewer and Nash (1989), this set was called the set of all objects, where an object was understood as a dataset of a company. In that paper, BN considered the binary relation induced by the conflict of interests among company dataset O; we will call it Conflict of Interest binary Relation (CIR). BN assumed O could be partitioned (decomposed) into mutually disjoint CIR-classes. Such a structure is called a partition in mathematics (Brualdi, 1992) and is well observed that a partition induces an equivalence relation and vice versa. Hence BN's assumption implies that CIR is an equivalence relation. In this section, we will discuss the "common sense" of CIR. Mathematically, CIR is a binary relation. The key question is: what properties does CIR possess?

Could CIR be reflexive? In other words, is it reasonable that every company self conflict? The answer is an obvious no.

Could CIR be symmetric? This is an obvious yes.

Could CIR be transitive? To discuss this one, let us look at a specific example. Let O = {USA, UK, USSR}, and let CIR be read as "has conflict with". If CIR were transitive, then the following two statements

> USA has conflict with USSR (so both are in the same CIR-class, say A)
> USSR has conflict with UK (so both are in the same CIR-class, say B)

would imply that

> USA has conflict with UK (A and B are the same class).

Obviously, the last statement is absurd. Though this is a very specific example, it is clear that the arguments have their generality: Assume X and Y are "friends", and both have conflict of interests with Z. That is, X and Z belong to the same CIR-class. By the same reasoning, both Y and Z should also be in a CIR-class. On the other hand, since X and Y are "friends", they cannot be in the same CIR-class, by definition. These two CIR-classes are distinct (since X and Y cannot be in the same class) but overlapped (Z is in both classes); so CIR cannot be transitive.

Main Thrust of the Chapter

In spite of their technical error, BN's intention (requirements) was a fascinating one. To keep their main intent, Lin (1989b) reformulated the model based on the recognition that "conflict of interests" is an "anti-equivalence-relation"; however, at that time, the expected sharpness and crispness of the model had not been established. With the recent development of granular computing, we can recapture the sharpness and crispness of the model. In other words, we will show that the *refined ACWSP does capture BN's intuitive idea and is indeed the correct model of the intended Chinese Wall Security Policy (CWSP) Model.*

Main Theorems: Establishing BN's Intended Model

In Lin (1989b), we introduced Aggressive Chinese Wall Security Policy (ACWSP) model that remedies the flaw of BN theory. In this chapter, we will refine ACWSP model and show that indeed it has BN's intended properties. Let O be the set of corporate data and CIR $\subseteq$ O×O be a binary relation that represents the conflict of interests.

Definition. The pair (O, CIR) is called an Aggressive Chinese Wall Security Policy (ACWSP) Model if O is the set of corporate data and CIR is anti-reflexive, symmetric, and anti-transitive.

Anti-reflexive and anti-transitive are extreme forms of non-reflexive and non-transitive. Non-reflexive means there are objects that are not reflexive, while *anti-reflexive means no objects are self-conflicting* in this application. Similarly, non-transitive means there is no transitive case, while anti-transitive has absolutely no transitive phenomena at all. The term *aggressive* means the Chinese Wall has encircled a maximal area.

Strong ACWSP Theorem. Let (O, CIR) be ACWSP model; then there is no information flow between X and Y if and only if (X, Y) $\in$ CIR, where CIR satisfies the three axioms.

Note that without the three axioms, Strong ACWSP Theorem is not true. Let IAR be the complement of CIR.

Complement Theorem. IAR is an equivalence relation if CIR satisfies the three axioms.

Both theorems will be proven in the Granular Computing section. In the next section, we will show that two theorems do imply that ACWSP Model does meet BN's requirements.

BN's Requirements

Let us set up some notation. Let DIF $\subseteq$ O×O be the binary relation of direct information flow. So, a direct information flow from X to Y is denoted by (X, Y) $\in$ DIF. Note that we will treat X and Y symmetrically, so *DIF is a symmetric binary relation.* The main idea of BN is to capture a correct intuition that is behind the term *conflict of interests*. Brewer and Nash (1989), in the section, "Simple Security", asserted that "people are only allowed access to information which is not held to conflict with any other information that they already possess" (p. 207).

1. This assertion does *not* allow an agent to read both datasets that have conflict of interests and allows an agent to read any non-conflicting datasets. Figuratively speaking, there is a Chinese Wall between any two conflicting datasets, and no information can penetrate through these Chinese Walls. Conversely, there are information flows (two directions) if two datasets are not in conflict.
2. This requirement implies that an agent is permitted and only permitted to read any non-conflicting datasets. Let us deliberate a little bit further: When the agent, who has read two datasets, say X and Y, is considering a decision for Y data, the information about X data may be used consciously or unconsciously. So, the permissions of reading two datasets imply the information can flow between two datasets (two directions). So formally, BN's assertion implies:

Simple CWSP:

$$(X, Y) \notin DIF \text{ if and only if } (X, Y) \in CIR, \text{ or equivalently}$$
$$(X, Y) \in DIF \text{ if and only if } (X, Y) \notin CIR$$

Simple CWSP does not exclude the possibility of information flow X to Y *indirectly*. Let CIF (composite direct information flow from X to Y) represent a finite sequence of DIF's

$$(X, Y) \in CIF \text{ means } X = X_0 \Leftrightarrow X_1 \Leftrightarrow X_2 \qquad \Leftrightarrow X_n = Y,$$

where we write DIF by more "geometric" notation $\Leftrightarrow$.

Granular Computing: Binary Relations, Granulations, and Neighborhood Systems

We can formulate binary granulation in three forms. Let V be the universe (a Cantor set) of discourse.

Binary Relation (BR): A binary relation is a subset, $B \subseteq V \times V$, of the Cartesian product $V \times V$.

For each $p \in V$, we may associate a set $B_p = \{v \mid (p, v) \in B\}$, which is called a binary neighborhood or a granule. This association $p \rightarrow B_p$ is called a binary granulation (BG). The collection $\{B_p \mid p \in V\}$ is called a binary neighborhood system (BNS). The notion of binary granulation and binary neighborhood system can be defined directly without referring to binary relation.

Binary Granulation (BG) and **Binary Neighborhood System (BNS):** A binary granulation is a mapping (single valued function),

$$V \rightarrow 2^V$$

where 2^V is the power set of V; the image N_p of p is called a granule at p or a binary neighborhood at p; and it could be an empty set. Then the collection

$$\{N_p \mid N_p \in 2^V \text{ for all } p \in V\}$$

is called a binary neighborhood system.

Neighborhood system (NS): If the image of the mapping above is, instead of a single element in the power set, *a subset* of 2^V, then the association is a generalization of the neighborhood system (TNS) of a topological space and is called a neighborhood system (NS).

Proposition: Binary relation (BR), granulation (BG), and neighborhood system (BNS) are equivalent concepts.

So, we will treat these terms as synonyms. The notation B may mean BR, BG, or BNS; B(p) is a granule/binary neighborhood.

The Induced Partition and the Center

The inverse map of binary granulation induces a partiton on V and will be called the induced partition/equivalence relation of B and denoted by C and called the center of B (Lin, 1998a, b).

Theorem. Let C be induced equivalence relation of a symmetric binary relation B. Then, each B-binary neighborhood is a union of C-equivalence classes. In such a situation, we say B is C-definable.

Proof: Let $x \in B_p$ be a point x in the binary neighborhood B_p. Let x and y be in the same C-equivalent class. By definition of equivalency, $B_X = B_Y$. As B is symmetric, $x \in B_p$ implies $p \in B_X$ $(= B_Y)$. By symmetry again, $y \in B_p$. As y is arbitrary, we have proved $[x] \subset B_p$, that is, B_p contains the equivalence class [x] of any member x in B_p. QED.

This proposition is the best possible; there are examples to show that without symmetry, B may not be C-definable; [x] is often denoted as C_X.

Compliment

Theorem. If B is symmetric, anti-reflexive, and anti-symmetric, then B' is an equivalence relation.

Proof: Since B is anti-reflexive, B' is reflexive. B is symmetric; so is B'. Now we need to prove B' is transitive. To do so, we need to prove: $\forall$w, (u, w) and (w, v) are in B' imply (u, v) is in B'. We will use indirect proof: we assume, on the contrary, that (u, v) is not in B', that is, (u, v) belongs to B. By anti-transitive of B, $\forall$w (only one of (u, w) or (w, v) belongs to B); this is contradictory to the assumption that $\forall$w (both (u, w) and (w, v) belongs to B). So the assumption that (u, v) does not belong to B' leads to an error. So, (u, v) does belong to B'; this proves the transitivity of B'; QED.

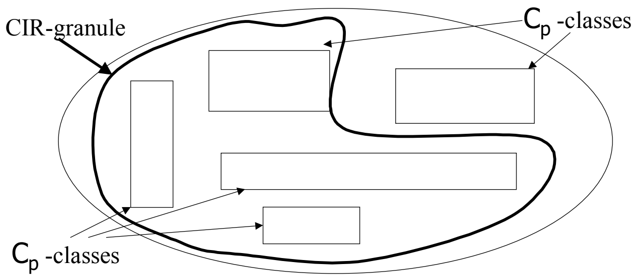
As a corollary, we have

Complement Theorem. IAR is an equivalence relation if CIR satisfies the three axioms.

Proposition. If B is symmetric, anti-reflexive, and anti-transitive, then B' is C.

Proof: Since B is symmetry, B_v is C-definable; that is, B_v is a union of C-equivalence classes. As B_v' is a complement of B_v, B_v'(v) should also be a union of C-equivalence classes. Hence C is a refinement of B'. Next, we only need to prove that B' is a refinement of C. For this purpose, we will prove that (v, u) $\in$ B' implies $B_v = B_u$ (that is, (v,u) $\in$ C). By previous theorem, B's assumptions imply that B' is equivalence relation. So, (v, u) $\in$ B' implies $B_v' = [v]_{B'} = [u]_{B'} = B_u'$. Note that B is the complement of B'; we have $B_v = B_u$. In other words, (v, u) $\in$ C.

Figure 1. Granulation and its center



The Proof of Strong ACWSP Theorem

We need some notations. Let X be an object (a company dataset) of O. Let CIR(X) and $[X]_{IAR}$ be the CIR-class (a binary granule) and IAR-class (equivalence class) of X respectively. To prove the theorems, we need to establish the following Strong ACWSP property:

There is CIF between X and Y if and only if $(X, Y) \notin$ CIR.

First, we will show "only if": We assume that there is CIF between X and Y; that is, there is a sequence of DIFs:

$$X=X_0 \Leftrightarrow X_1 \Leftrightarrow X_2 \cdot \cdots \Leftrightarrow X_n=Y$$

From the Complement Theorem, IAR is an equivalence relation, so we have

$$[X]_{IAR} = [X_0]_{IAR} = [X_1]_{IAR} = \ldots = [X_{(n-1)}]_{IAR} = [X_n]_{IAR}=[Y]_{IAR}$$

In other words, $(X, Y) \in$ IAR, hence, as CIR is the complement, we have $(X, Y) \notin$ CIR; this proves the "only if". To see the "if", it is clear from the right hand-sided assumption, $(X, Y) \notin$ CIR, we have a direct information flow from X to Y, which is a special CIF. Hence, we have the "if" part. Thus, we have established the Strong ACWSP Theorem.

Future Trends

The central notion of Discretionary Access Model (DAC) is that the owner of an object has discretionary authority on the access rights of that object. The owner X of the dataset x may grant the read access of x to a user Y who owns a dataset y. The user Y may make a copy, Copy-of-x, in y. Even in the strict DAC model, this is permissible (Osborn, Sandhu, & Munawer, 2002). We have summarized the above grant access procedure, including making a copy, as a direct information flow (DIF) from X, x to Y, or y, respectively. For convenience,

Convention: We will assume there is a virtual owner for every object(dataset); so the information flows among agents or object are the same. This simplifies the discussions; the assignment of a real human to a virtual agent is separated from flow analysis.

With this convention, we can focus on *a direct information flow (DIF) from X to Y*. In DAC model, each object X has an access list, that is, a list of virtual agents that can read the data X. Equivalently, each object, $X \in O$, is associated with a subset, $DIF(X) \in 2^O$, of objects that X can flow into. So formally, DAC is a 2-tuple,

$$(O, DIF)$$

where O is a collection of objects (datasets), and DIF is an amp

$$DIF: X\ (\in\ O) \to DIF(X)\ (\in\ 2^O)$$

that associates each object, X, a subset, DIF(X), of objects that X can flow into. In the language of granular computing, DAC has a binary granulation, and

DIF(X) is a binary neighborhood/granule at X. By the proposition in the Granular Computing section, DIF is a BG, BNS, and BR. In DAC language, the binary relation DIF has been referred to as direct information flow relation. In general, DIF is not symmetry; we denote it by $\Rightarrow$:

$(X, Y) \in$ DIF if and only if $X \Rightarrow Y$

In such a case, we say that a direct flow from X to Y is permissible. In conclusion, we say

Proposition. The discretionary access model, DAC, mathematically is a universe (a set) of objects, in which the universe is equipped with a binary granulation, a binary neighborhood system, or a binary relation.

One of the fundamental weak points of DAC is the Trojan horse problem. Our approach to CWSP provides some hints to the directions of general solutions. Let us review DAC through the eye of *simple CWSP. To understand its mathematical mechanics, we* shall formulate the *simple CWSP* model in a very general setting; namely, no assumptions are imposed on the binary relation of "conflict of interests." Formally, a DAC Model with simple CWSP is a pair (O, DIF), where

$(X, Y) \notin$ DIF if X and Y are in "conflict of interests"

Note that the right-handed side is a condition expressed in the language of application domain. So, an "abstract" simple CWSP is

$(X, Y) \notin$ DIF if X and Y meet *G*, where G is a general constraint on the application domain.

$(X, Y) \notin$ DIF is awkward; *we will use* explicitly denied access model. Let NDIF(X) be the explicit denied access list (Lunt, 1989). So, we have

$(X, Y) \in$ NDIF if and only if X and Y meet *G*.

Thus, the ACWSP Theorem is a consequence of appropriate properties of G (e.g., the three axioms). Without using domain condition, it is a consequence of NDIF's properties. Therefore, a general ACWSP theorem and complement Theorem for DAC can be stated as follows:

ACWSP Theorem for DAC. If NDIF is anti-reflexive, symmetric, and anti-transitive, then

1. **A CWSP Theorem:** $(X, Y) \in$ NDIF (e.g., $(X, Y) \in$ CIR) implies $(X, Y) \notin$ CIF and
2. **Compliment Theorem:** DIF (i.e., the compliment of NDIF) is an equivalence relation.

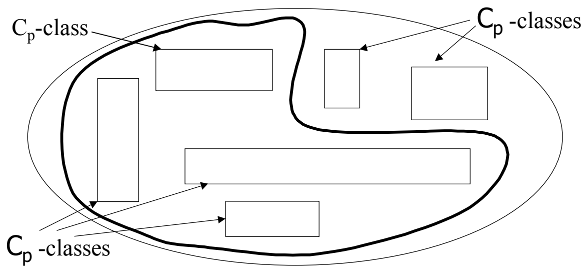
The compliment theorem is actually a Trojan horse problem: $(X, Y) \in$ NDIF means information flow from X to Y is impermissible. $(X, Y) \notin$ CIF means absolutely no information can flow from X to Y whether directly or indirectly. This is closely related to the Trojan horse problem

In ACWSP, BN require that an agent can read any datasets as long as there are no conflicts; this requirement is too strong. In security environment, the "need to know" is a very strong requirement, so no conflict does not imply the permission of read by one agent. In other words, the collection of no-conflict datasets is not necessarily an equivalence class. Let C_{NDIF} be the induced equivalence relation (from NDIF).

Trojan horse Theorem for DAC. If NDIF is anti-reflexive and symmetric, then the DAC (O, C_{NDIF}) has no malicious Trojan horses.

Since the access list C_{NDIF} is an equivalence relation, there is no Trojan horses; note that C_{NDIF} is the induced partition and not the complement of NDIF.

Figure 2. $C=C_{NDIF}$ is the center of B



Conclusions

Granular computing reveals the link between the binary relation of information flows and access list (binary granulation). Based on such a link, we settle the security of CWSP. In other words, granular computing provides techniques of regulating information flow (binary relation) through the "access list" (binary granulation). Granular computing will be a good theory in studying information flows in the DAC model.

The theorem in The Induced Partition and the Center section implies that if we take the "need to know" constraints into the Simple CWSP, we still can prove that Simple CWSP implies Aggressive CWSP. In this case, the compliment of CIR is not an equivalence relation, and an agent can only read the company datasets that have the same set of competing companies.

References

Bell, D. (1987). Secure computer systems: A network interpretation. *Proceedings of the 3rd Annual Computer Security Application Conference* (pp. 32-39).

Brewer, D.C., & Nash, M.J. (1989, May). The Chinese Wall security policy. *Proceedings of the IEEE Symposium on Security and Privacy* (pp. 206-214).

Brualdi, R.A. (1992). *Introductory combinatorics.* Prentice Hall.

Chu, W., & Chen, Q. (1992). Neighborhood and associative query answering. *Journal of Intelligent Information Systems, 1*, 355-382.

Demurjian, S.A., & Hsiao, S.A. (1988, August). The multimodel and multilingual database systems: A paradigm for the studying of database systems. *IEEE Transaction on Software Engineering, 14*(8).

Hsiao, D.K., & Harary, F. (1970, February). Formal system for information retrieval from files. *Communications of the ACM, 13*(2).

Lee, T.T. (1983, December). Algebraic theory of relational databases. *The Bell System Technical Journal, 62*(10), 3159-3204

Lin, T.Y. (1988, February). Neighborhood systems and relational database. *Proceedings of the 1988 ACM 16th Annual Computer Science Conference* (pp. 725-X).

Lin, T.Y. (1989a, October). Neighborhood systems and approximation in database and knowledge base systems. *Proceedings of the 4th International Symposium on Methodologies of Intelligent Systems* (pp. 75-86).

Lin, T.Y. (1989b, December). Chinese Wall security policy: An aggressive model. *Proceedings of the 5th Aerospace Computer Security Application Conference* (pp. 282-289).

Lin, T.Y. (1992, September). *Attribute based data model and polyinstantiation. Education and Society,* IFIP-Transaction, ed. Aiken, 12th Computer World Congress, 472-478.

Lin, T.Y. (1996, September). Rough logic formalism for fuzzy controllers: A hard and soft computing approach. *Journal of Approximate Reasoning, 15*(4), 395-414.

Lin, T.Y. (1997, September). From rough sets and neighborhood systems to information granulation and computing in words. *European Congress on Intelligent Techniques and Soft Computing,* 1602-1606

Lin, T.Y. (1998a). Granular computing on binary relations I: Data mining and neighborhood systems. In A. Skoworn, & L. Polkowski (Eds.), *Rough sets in knowledge discovery* (pp. 121-140). Physica-Verlag.

Lin T.Y. (1998b). Granular computing on binary relations II: Rough set representations and belief functions. In A. Skoworn, & L. Polkowski (Eds.), *Rough sets in knowledge discovery* (pp. 121-140). Physica-Verlag.

Lin, T.Y. (2002a, August). Placing the Chinese Walls on the boundary of conflicts: Analysis of symmetric binary relations. *Proceedings of the International Conference on Computer Software and Applications* (pp. 966-971).

Lin, T.Y. (2002b, October). Granular computing on binary relations: Analysis of conflict and Chinese Wall security policy. *Lecture Notes on Artificial Intelligence, 2475*, 296-299

Lin, T.Y. (2002c). Attribute (feature) completion: The theory of Attributes from data mining prospect. *Proceedings of the IEEE International Conference on Data Mining*, December 9-12, Maebashi, Japan.

Lin, T.Y., & Chiang, I.-J. (2004). Automatic document clustering of concept hypergraph decomposition. *Proceedings of SPIE,* April 12-13, Orlando, FL.

Lin, T.Y., & Chiang, I.-J. (2005). A simplicial complex, a hypergraph, structure in the latent semantic space of document clustering. *Journal of Approximate Reasoning*.

Liu , C.H., Chiang, I.-J., Wong, J.-M., & Lin, T.Y. (2004). A complex bio-network of the function profile of genes. *Proceedings of the Foundation of Data Minig* (pp. 107-114).

Lunt, T.F. (1989). Access control for database systems, database security II: Status and prospects. *Proceedings of the IFIP WG11.3 Workshop on Database Security*, October 5-7, Kingston, Canada.

Motro, A. (1986). Supportin gaol queries. *Proceedings of the 1st Internatonal Conference on Expert Database Systems,* April (pp. 85-96).

Osborn, S., Sandhu, R., & Munawer, Q. (2002, May). Configuring role-based access control to enforce mandatory and discretionary access control policies. *ACM Transactions on Information and Systems Security, 3*(2), 85-106.

Pawlak, Z. (1982). Rough sets. *International Journal of Information and Computer Science, 11*, 341-356.

Pawlak, Z. (1984). On conflicts. *International Journal of Man-Machine Studies, 21*, 127-134.

Wong, E., & Chiang, T.C. (1971, September). Canonical structure in attribute-based file organization. *Communications of the ACM, 14*(9).

Zadeh, L.A. (1979). Fuzzy sets and information granularity. In M. Gupta, R. Ragade, & R. Yager (Eds.), *Advances in fuzzy set theory and applications* (pp. 3-18). Amsterdam: North-Holland.

Zadeh, L.A.(1998). Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/ intelligent systems. In *Soft computing 2* (pp. 23-25). Springer-Verlag.

Zadeh, L.A.(2002). Some reflections on information granulation and its centrality in granular computing, computing with words, the computational theory of perceptions and precisiated natural language. In T.Y. Lin, Y.Y. Yao, & L. Zadeh (Eds.), *Data mining, rough sets, and granular computing*.

Section III

Security for Emerging Applications

Chapter XI

A Multimedia-Based Threat Management and Information Security Framework

James B. D. Joshi, University of Pittsburgh, USA

Mei-Ling Shyu, University of Miami, USA

Shu-Ching Chen, Florida International University, USA

Walid Aref, Purdue University, USA

Arif Ghafoor, Purdue University, USA

Abstract

This chapter focuses on the key challenges in the design of multimedia-based scalable techniques for threat management and security of information infrastructures. It brings together several multimedia technologies and presents a conceptual architectural framework for an open, secure distributed multimedia application that is composed of multiple domains employing different security and privacy policies and various data analysis and mining tools for extracting sensitive information. The challenge is to integrate such disparate components to enable large-

scale multimedia applications and provide a mechanism for threat management. The proposed framework provides a holistic solution for large-scale distributed multi-domain multimedia application environments.

Introduction

Security of information infrastructures, both in public or private sectors, is vital to overall national security goals. Such infrastructures provide capabilities for gathering, managing, and sharing vital information among numerous organizations that can form large e-enterprises and generally interoperate in the form of a federation of autonomous domains (Joshi, Ghafoor, Aref, & Spafford, 2001; Thuraisingham, 2003). Information shared among multiple domains can come in various forms including text, audio, video, and images which can increase the complexity of security and privacy management. The key security challenges include integration of diverse security policies of collaborating organizations into a coherent capability for protecting information and using collaborative knowledge for detecting and responding to any emerging threats. In addition, information privacy is generally an overriding concern (Adams & Sasse, 1999). Furthermore, a plethora of data analysis and mining tools have emerged that cyber defenders can use to extract sensitive information from public and private multimedia applications and detect patterns and activities indicating potential threats to an infrastructure. Thus, two key challenges to the design of multimedia-based scalable techniques for threat management and security of information infrastructures are *data mining* and *security*, which we briefly overview in the next section.

Key Issues in Data Mining and Multimedia Security

Multimedia Data Analysis and Mining

Emerging multimedia applications require large-scale integration, mining, and analysis of multimedia data that is generally distributed over multiple security domains. Most of these applications use sensitive information for identifying complex threat actions that cannot be detected via real-time monitoring as such actions can take place over relatively long timeframes. Examples of such applications include detecting the spread of an epidemic and monitoring deterioration of the environment. However, today, data no longer appears in the text form only. Instead, the information from different sources may be in the form of text, image, video, audio, or multimedia documents consisting of several multimedia objects that are tightly synchronized both in space and time (Little & Ghafoor, 1990). Unlike mining the relational data, multimedia data mining is a more complex issue due to the sheer volume and heterogeneous characteristics of the data and the spatial and/or temporal relationships that may exist among multimedia data objects.

Mining multimedia data has recently been addressed in the literature (Chen et al., 2003a; Chen et al., 2004; Thuraisingham, 2003). Most of the existing approaches, however, provide limited capabilities in terms of content analysis and generally do not exploit correlations of multiple data modalities originating from diverse sources and/or sensors. Real-time mining and correlating of multi-modality data from distributed sources and using security-oriented spatio-temporal knowledge can assist in identifying potential threats and ensuring security of large-scale infrastructures (e.g., in command and control environments). In a broader perspective, both *long-ranged* and *real-time* data analysis and mining techniques are needed to allow multi-level content analysis and representation of multimedia data at different levels of resolution to facilitate information classification that has security and privacy implications.

Security Policy and Privacy Management

The multi-modality nature of data and the unique synchronization and *quality of service* (QoS) requirements of multimedia information systems makes its protection uniquely challenging. These challenges are briefly discussed below.

Content-based Context-aware Access. An application may require controlled access to information based on the sensitivity level of information content, time, location, and other contextual information obtained at the time the access requests are made. For example, in the health care industry, selective content-based access to patient information should be given to physicians, insurance providers, and so forth. Furthermore, a non-primary physician of a patient, who normally does not have access to the patient's records, may need to be allowed access in a *life-threatening* situation.

Heterogeneity of Subjects and Objects. For multimedia applications, heterogeneity implies the diversity of object and subject types. Object heterogeneity implies different types of media, abstract concepts, or knowledge embodied in the information that needs to be protected. For instance, in a digital library system, it is desirable to provide access based on concepts rather than on individual objects. For example, a user may want to access information related to concept *Juvenile Law*. Subject heterogeneity implies that the users have diverse activity profiles, characteristics, and/or qualifications that may not be known *a priori*, which can complicate the specification of access control requirements.

- **Privacy:** Because of the richness of information content of multimedia data, privacy concerns are exacerbated in a multimedia environment (Adams & Sasse, 1999). For instance, a perfectly harmless video sequence of a conference presentation may show a certain behavior of a person (e.g., sleeping during the presentation) in the audience which may result in adversely affecting his/her status if it is viewed by his employer.
- **Dynamically Changing Access/Privacy Requirements:** Access control and privacy requirements in Web-based multimedia applications are inherently dynamic in nature, as the information contents may change frequently. Powerful and flexible access control and privacy models are needed to capture dynamically changing access and privacy parameters in an evolving system.
- **Integration of Access and Privacy Policies:** As multimedia information often exists in distributed heterogeneous administrative domains, there is a growing challenge of developing efficient integration and evolution management tools and techniques for inter-domain access control policies for distributed multimedia applications. Mechanisms are required to support careful analysis of security policies for the multi-domain environment and for mapping multimedia content parameters associated with the *security clusters* that data mining tools may generate to define content-based access control and privacy policies. Situations can also arise when the existing static security policies, both local and global, may not provide sufficient safeguards against emerging threat

scenarios and require changes as a scenario unfolds based on the information extracted from newly mined data. In such situations, real-time analysis of dynamic access control policies is required to prevent any potential security breaches in multimedia applications. In such an integrated environment, another key challenge is to efficiently manage the evolution of local and global information classifiers, and security and privacy policies.

Central to these problems is a crucial issue of uniform representation, interchange, sharing, and dissemination of information content over an open, multi-domain environment. Due to the multi-modality nature of multimedia data, the real-time constraints, and heterogeneity of distributed domains, an integrated solution is needed to address the above-mentioned data mining, and security and privacy challenges. To meet these requirements, the integration of several technologies such as data mining, distributed multimedia systems, access control, and privacy models and mechanisms is needed. *EXtensible Markup Language* (XML) (Bertino, Castano, Ferrari, & Mesiti, 1999; Damiani, di Vimercati, Paraboschi, & Samarati, 2002) has emerged as a promising technology for addressing such a fundamental issue of information representation, interchange, sharing, and dissemination (Thuraisingham, Clifton, & Maurer, 2001). XML supports a rich set of features including user-defined tags, nested document structures, and document schemata (Bertino et al., 1999; Bertino, Castano, & Ferrari, 2001; Damiani et al., 2002). Several emerging XML-related technologies, including *Resource Definition Framework* (RDF), *DARPA Agent Markup Language + Ontology Inference Layer* (DAML+OIL), and *Web Ontology Language* (OWL), provide support for integrating information from different domains by facilitating semantic matching of elements (Hunter, 2003; McGuinness, Fikes, Hendler, & Stein, 2002). Moving Pictures Experts Group's MPEG-21 is an open multimedia standard that supports multimedia content delivery and consumption (Burnett et al., 2003). MPEG-21, however, does not capture flexible access control requirements outlined earlier. To the best of our knowledge, no prior research has addressed the above issues in a unified manner by integrating various technologies for developing a secure distributed multi-domain multimedia environment, although several researchers have addressed some of these issues. In this chapter, we discuss various challenges and present a conceptual framework for a scalable secure multi-domain multimedia information system that addresses these challenges.

Conceptual System Architecture

Various functional components of the conceptual architecture for a single domain multimedia environment are depicted in Figure 1. The *XML-based Multimedia Document Composition Module* (XDCM) provides an interface for composing XML schemata for multimedia information and policy documents and includes *Policy Composition Interface*. Access control and privacy policy documents are stored in the *XML Policy Base* (XPB). The *Access Control and Privacy Enforcement Modules* (ACM, PEM) constitute the key enforcement modules. The ACM employs a content-based context-aware RBAC engine, extracts the policy information from the XPB, and interacts closely with the *XML View Generator* (XVG) module to produce the authorized presentation schema for a user. Authorized XML instances are further checked against privacy policies by the PEM to ensure that privacy policies are enforced. The *Session Management Module* (SMM) is responsible for monitoring the real-time activities of the multimedia presentation, as well as dynamic access constraints, and interacts with ACM and XVG to update active XML views.

Figure 1. Conceptual design of a single domain multimedia application environment

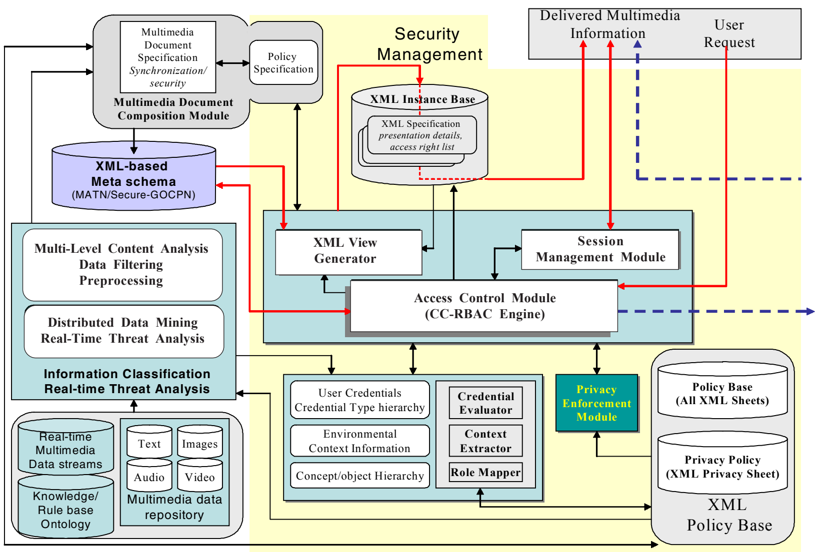
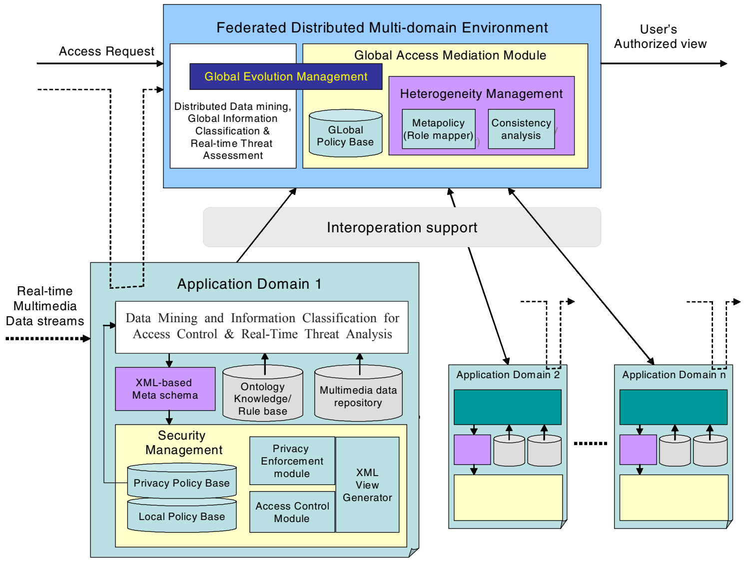


Figure 2. Conceptual design of a multi-domain multimedia application environment



The *Information Classification and Real-time Threat Analysis Module* (ICRTAM) is responsible for the classification of multimedia data and threat analysis based on real-time multimedia data streams. The clusters that it generates are organized in a hierarchy. The *Role Mapper* generates concept roles for these clusters and creates required policy sheets that define access rules and privacy protection requirements. The *Credential Evaluator* (CEv) module evaluates the credentials presented by the ACM. With the help of the *Role Mapper*, it maps the credentials to a role as defined by the access rules. The *Context Extractor* (CEx) evaluates the contextual information and sends the results to the ACM so it can evaluate context-based dynamic policies. The multimedia data repository constitutes the physical objects present in the system that are used to generate the XML multimedia object *meta-schema*, by integrating multimedia presentation information with security and privacy attributes. The *knowledge/rule base* or the *ontology* will be used by the information classification and security cluster generator module.

The *Multi-level Content Analysis, Data Filtering and Pre-processing Module* (MCADFPM) is responsible for (1) conducting multi-level content analysis for mining and learning the semantic information; (2) exploiting suitable multi-resolution representation schemes for various media features, segments, and objects; and (3) enabling automatic feature selection to reach the compact sets of features and dataset for data mining in the data-filtering and preprocessing step. The module assists in generating the security clusters and their attribute and feature sets. The *Distributed Data Mining and Real-time Threat Analysis Module* (DDMRTAM) is responsible for designing and developing (1) distributed data mining techniques for threat assessment based on real-time streams and (2) an information fusion model to resolve the inconsistencies when integrating the classification results from multiple local classifiers. The information clusters that it generates are organized in a hierarchy and represented in an XML.

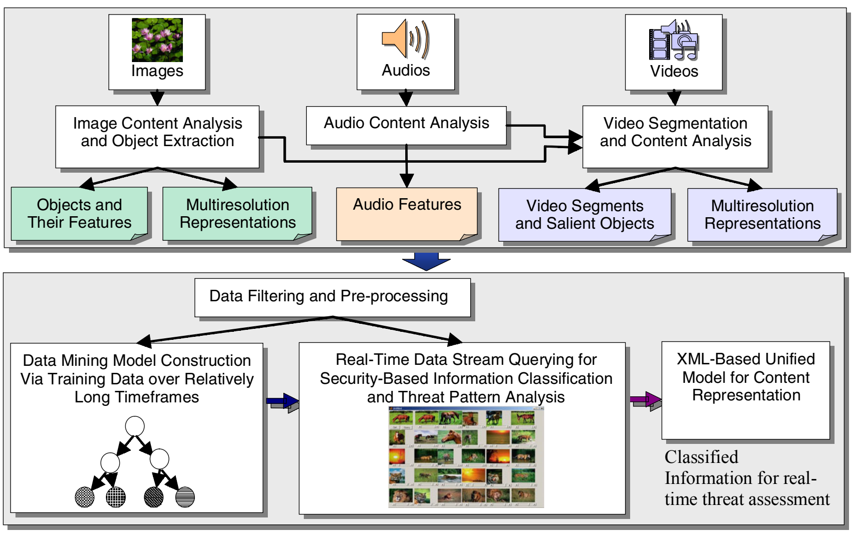
Conceptual System Architecture for a Multi-Domain Multimedia System

Figure 2 depicts the key components of the conceptual architecture of a secure federated distributed, multi-domain multimedia environment. The *Distributed Data Mining, Global Information Classification and Threat Assessment* component maintains global level views of conceptual objects based on the security clusters of individual domains. It also allows inter-site conceptual mapping of information objects and distributes real-time threat assessment tasks with the help of the ontology or knowledge base of each domain. The *Global Access Mediation Module* integrates all the local policies into a unified *Global Policy Base*. The *Heterogeneity Management* module resolves any semantic differences and inconsistencies among the local policies when the global policy is constructed using the *Consistency Analysis* module. The final global level policy generated for the whole multi-domain environment primarily consists of a *Role Mapper* that maintains the mapping from global roles to local roles for access mediation. For a global access request, these mappings are used to issue access requests to individual domains. *Global Evolution Management* module is responsible for overseeing the process of ensuring consistent evolution of local information classification mechanisms and security/privacy policies.

Scalable Multimedia Data Analysis and Mining Techniques

Content-based analysis of multimedia information for facilitating distributed security policies is a challenging issue because of the multi-modality nature of multimedia data and heterogeneity of distributed information sources and sensors. Scalable real-time content-based analysis and mining frameworks are needed to generate security-oriented classification of multimedia information for the distributed multi-domain environment and to provide capabilities for real-time threat assessment. An efficient approach of real-time data stream querying across multiple domains needs to be developed. A set of classifiers, both for individual domains and for global environment, that use a set of training data over relatively long timeframes is essential. Based on the semantics and the contents of information, these classifiers will produce a set of security clusters of multimedia objects, each with a set of features and attributes that can be used for developing information security and privacy policies for single and multi-domain environments, as discussed later. Another key challenge for designing the global classifier for multiple domains is to fuse information from local

Figure 3. Architecture of multi-modal content data analysis and real-time data mining



classifiers and resolve any semantic conflicts and inconsistencies in an efficient manner. Figure 3 depicts the data analysis and classification processes involved in multimedia information systems.

Real-Time Content Analysis of Multi-Modality Multimedia Data. An in-depth content analysis and mining of multimedia data can provide a tremendous wealth of information useful for security management and threat assessment. For this purpose, multi-modality content analysis is essential for integrating and transforming multimedia data into a format suitable for off-line data mining model construction and development of security clusters. Such analysis can span relatively long timeframes. On the other hand, data stream querying for classification and threat assessment requires real-time processing.

Multi-Level Content Analysis of Multi-Modality Data. To automate multi-level real-time content analysis of multimedia data, a framework is needed that allows:

1. extraction of low-level features, including the color and shape features of an image or video frame and the energy feature of an audio segment;
2. extraction of mid-level features such as the object segmentation based on statistical features of the low-level features and the video shot/scene segmentation;
3. extraction of high-level features that identify high-level multimedia content or events; and
4. incremental learning to adapt to the changes in semantics over time.

Mining single-modality alone is not enough to reveal and interpolate the complex correlations of multiple data modalities from diverse sources. Therefore, it is more desirable to discover knowledge from multimedia data with multi-modalities and temporal relationships (Chen et al., 2003; Chen et al., 2004; Dagtas & Abdel-Mottaleb, 2001). A conceptually simple problem in information fusion is the combination of *opinions* of multiple classifiers. Several information fusion frameworks have been proposed, including the *Bayesian model*, *majority voting rule*, *average of classifier outputs*, *Naïve-Bayes*, *decision template*, and *dynamic classifier selection* (DCS) for integrating information from sources with different modalities (Giacinto, Roli, & Didaci, 2003; Krzysztofowicz & Long, 1990).

Representation of Multimedia Data. Conventional multimedia database systems represent multimedia objects at the single level resolution. Multi-resolution representation consists of maintaining multiple and concurrent representation of multimedia objects and plays an important role in providing semantics to support access control and privacy of information. For example, a confidential medical X-ray image may also have the patient identity information. However, only the version without the identity information should be displayed to unauthorized users. Multi-resolution representation schemes for static images and video data based on the multi-splines method have been proposed in Moni and Kashyap (1995), which has shown to yield excellent visual image quality. In addition, multi-resolution representation provides an attractive solution in terms of coding complexity and image quality.

Real-Time Security-Based Information Classification and Threat Assessment. A set of suitable data mining classifiers, both for individual domains and for the distributed multi-domain environment, are required to support a comprehensive security framework. Such classifiers can be constructed off-line using the training data over relatively long timeframes and used to generate information categorization for implementing the underlying content-based access control policies with a wide range of classification parameters such as sensitivity level of information, its importance, and its semantics. In addition, real-time classifiers are needed to perform threat pattern analysis for detecting any emerging threats. The global classifier will integrate all the local classifiers and resolve the inconsistencies of the information among the local classifiers. These classifiers can generate security clusters consistent with access control and privacy management policies. Steps involved in the information classification process are discussed next.

Data Filtering and Pre-processing. The most challenging step prior to carrying data mining is to clean and prepare the data via data cleaning tools and/or domain-specific security knowledge or rules to ensure that the inputs of the data mining models are consistent with the overall security goals. The lack of a guide to choose the most appropriate subfeature set can severely limit the effectiveness and efficiency of data pre-processing. Moreover, in cases when security knowledge is not available, selecting a subset of compact features can become a bottleneck, especially when the features set is large. Several research works address this issue (Chen et al., 2003; Dy & Brodley, 2000; Shyu et al., 2003). A challenge is to develop efficient data cleaning and prefiltering techniques by enabling automatic security and privacy policy driven feature selection and eliminating unnecessary data records with the aid of domain knowledge.

Real-time Content Correlation of Multiple Streams for Event Detection. There exist no scalable mechanisms that can automatically correlate and synchronize multiple streams in multi-modalities and in spatial/temporal dimensions to support real-time data stream querying and event detection. Recently, a technique based on event-template for mining event information by combining audio and video cues to discover high level events in a medical database, such as surgery and patient-physician dialogs in the context of a video database management system was proposed (Aref et al., 2003; Hammad, Franklin, Aref, & Elmagarmid, 2003; Sandhu, Coyne, Feinstein, & Youman, 1996; Zhu et al., 2003).

Off-line Data Mining Model Construction. Given the training data generated from the data filtering and pre-processing steps, another challenge is to develop a suitable, scalable data mining classifier that can generate relevant security clusters and their feature and attribute sets for implementing access control policies and identifying threat patterns. A data mining classifier is constructed off-line by using the training data over relatively long timeframes. Generally, such classifiers can be used not only in evolving the existing threat patterns but also in discovering new threat patterns, which is extremely vital in real-time threat assessment. Conflicts and inconsistencies may arise while integrating information from multiple classifiers belonging to different domains into a global classifier, requiring information fusion to resolve these problems. A number of fusion rules can be used to combine an individual classification generated by training local classifiers.

XML-based Unified Model for Content Representation. The security clusters identified by the data mining step will consist of complex multimedia objects, which usually come from multiple application domains and in multi-modalities; hence, a unified model for content representation is needed. One promising approach is to use XML for multimedia content representation and develop an XML-based meta-schema for the security clusters and their attribute and feature sets so that these clusters and their sets can be directly used for developing the security management component. In addition, the XML-based meta-schema will represent the multimedia composition schema for secure presentation instances that may be composed of text, audio, video, and images. XML-based meta-schema is also used to represent the information retrieved by the data mining process and to identify multimedia documents and objects corresponding to the features and classes that the classifiers use, forming a high level conceptual "document."

Distributed Multimedia Security and Privacy Management

The development of a fine-grained, content and context based access control and privacy model for multimedia applications raises a number of serious technical challenges. The privacy issue is a major concern in multimedia systems. Efficient techniques are required to generate correct mapping between the security and privacy requirements and the data analysis and classification mechanisms that generate security clusters. A unified XML-based language for expressing the security and privacy policies, as well as complex synchronization constraints, can provide a basis for wider acceptance for practical use.

Access Control and Privacy Protection for Multimedia Information Systems

Access control may need to be applied at different levels of granularity and over hierarchically arranged multimedia objects. Few existing access control models address specific requirements of multimedia information systems. Bertino, Hammad, Aref, and Elmagarmid (2000) proposed a video database access control model that allows access control at various levels of granularity of video information, including video frames, video objects, or a sequence. Joshi, Fahmi, Shafiq, and Ghafoor (2002) proposed a *Secure Generalized Object Composition Petri-net* (Secure-GOCPN) model by extending the GOCPN model of multimedia documents to support multi-level security policies by classifying multimedia information into a prespecified set of classification levels. Subjects are also assigned security clearances that allow them to view information at or below a particular classification level. The unclassified level may correspond to preventing certain frames from being viewed by an unclassified user or hiding sensitive information from each frame. Using the mapping of classification levels across multiple domains, an implementation architecture that supports the presentation of distributed multimedia documents that include media documents/objects from multiple sources was presented in Joshi et al. (2002). The Secure-GOCPN model, however, only supports multi-level security policy. We propose using a role-based access control (RBAC) approach to address the generic security and privacy requirements of multimedia applications. RBAC supports fine-grained separation of duty (SoD) constraints, principle of least privilege, and efficient permission management. It also facilitates easy integration of policies because of the similarity of roles in different application domains (Ahn & Sandhu, 2000; Ferraiolo et al., 2001; Joshi et al., 2001; Kobsa & Schrek, 2003; Sandhu, Ferraiolo, & Kuhn, 2000). Atluri, Adam, Gomaa, and Adiwijaya (2003) proposed a preliminary formal model of access control of multimedia information by enhancing Synchronized Multimedia Integration Language (SMIL) to express policies.

A Parameterized RBAC Approach to Multimedia Information Systems. In an open environment such as the Web, a system should be able to facilitate a dynamic association of unknown users to a set of authorized actions. A content-based, context-aware RBAC framework would allow the users to be mapped to the activities authorized for them, based on context and credential information they present when requesting accesses. Roles also provide an abstraction level that can be used to capture security relevant features of multimedia objects and the security clusters generated by data mining tools. To handle these issues, one approach is to use parameterized roles and role preconditions that will test the attribute and feature sets of the security clusters and multimedia objects being accessed, the credentials of the requester, and the contextual information that should be valid at the time of access. Along this direction, significant work has been done in the temporal RBAC (TRBAC) model by Bertino, Bonatti, and Ferrari (2001) and later in the generalized TRBAC (GTRBAC) model by Joshi, Bertino, Latif, and Ghafoor (2005). Depending on the application semantics, not all roles may be available to all users at any time. This notion is reflected in the GTRBAC model by defining the following states of a role: *disabled*, *enabled*, and *active*. The *disabled* state indicates the role cannot be activated in a session. The *enabled* state indicates that authorized users activate the role. A role in the *active* state implies that there is at least one user who has activated the role. Accordingly, the preconditions that change the states of a role are: (1) *role enabling/disabling precondition* to enable/disable a role; (2) *role assignment/deassignment precondition* to assign/deassign a user (or permission) to the role; and (3) *role activation/deactivation precondition* that specifies an authorized user can activate a role or when a role should be deactivated.

Though several researchers have addressed access control using parameterized roles, no formal model has been presented by any. One approach is to extend the event-based GTRBAC model to develop a comprehensive event and rule-based, parameterized RBAC framework. A preliminary version of a parameterized role is presented in Joshi, Bhatti, Bertino, and Ghafoor (2004), which essentially builds on the event-based constraint framework of the GTRBAC model.

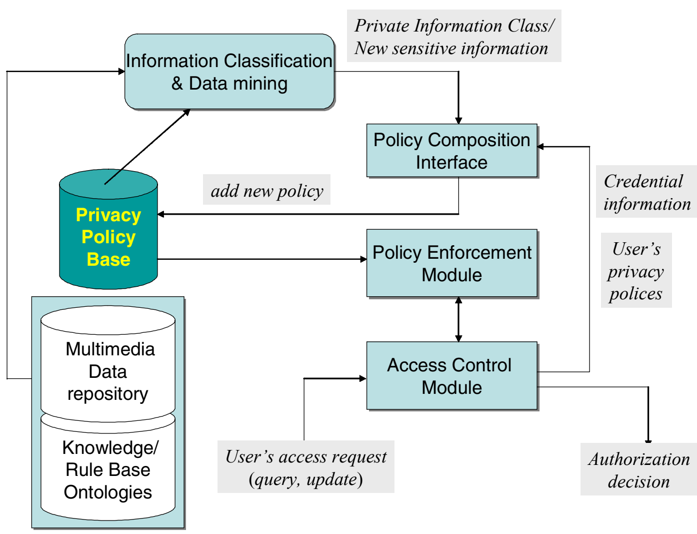
Extension of an XML Language for CC-RBAC Policies. We propose using an XML language that can express CC-RBAC modeling entities such as user credentials, role events and preconditions, clusters and multimedia objects, and permissions associated with them by adopting the X-RBAC language proposed by Joshi et al. (2004). A general credential expression of the form (*credTypeID*, *attributes*), where *credTypeID* is a credential type identifier and *attributes* is a set of attribute-value pairs, is used to map unknown users to predefined roles. In Joshi et al. (2004), an XML language for the basic RBAC model has been proposed to support parameterized RBAC policies. X-RBAC has been developed specifically for protecting XML content —*schema*, *instances*, and *elements*. The *schema*, *instances*, and *elements* of XML documents can naturally be extended to address the requirements of multimedia documents. Based on access control defined for the security clusters of multimedia objects and the presentation schema for such objects, presentation views for the users can be generated. Conceptual level access control policies use concept roles (Bhatti, Joshi, Bertino, & Ghafoor, 2003). It is possible that a schema element does not belong to the cluster to which its parent element belongs. It may further be necessary to specify that the element is not protected by a schema-level policy. At the lowest granularity, the protection may be applied to the fragments of an object such as on portions of a video sequence and individual frames. The XML document structure for the specification of multimedia information also demands a specific set of privileges such as authoring a part of the XML document, modifying attributes only of a particular subset of the XML specification of a multimedia presentation, and navigating privileges to allow the use of links to navigate linked multimedia information.

A considerable amount of work on RBAC can be found in the literature (Ahn & Sandhu, 2000; Ferraiolo et al., 2001; Sandhu et al., 2000). Giuri and Iglio (1997) presented the concepts of parameterized privileges and role templates to allow the specification of policies based on the content of the objects. In Georgiadis, Mavridis, Pangalos, and Thomas (2001), the context of a team is composed of user context and object context. None of these studies addresses the consistency issues and complex requirements of multimedia applications. The environment roles in Covington et al. (2001) are essentially statically defined context roles and thus cannot be employed in a dynamic environment. In Adam, Atluri, and Bertino (2002), a content-based access control (CBAC) model for a digital library has been proposed to protect the textual content of digital archives, uses credentials, and concept hierarchies. Related work also includes several security mechanisms for schema and XML-instance level protection of XML documents (Bertino et al., 1999; Damiani et al., 2002). Qin and Atluri (2003) recently proposed a content-based access control model for Semantic Web and used the concept level protection. These studies, however, do not address the security challenges in emerging multimedia applications mentioned earlier using an RBAC approach.

Privacy Protection in Multimedia Information Systems. In multimedia information systems, privacy concerns exist at multiple layers (Adams & Sasse, 1999). At a primary level, privacy relates to the sensitivity of actual information content of the multi-modal data. At the secondary level, it relates more to the sociopsychological characteristics of the data being accessed (Adams & Sasse, 1999). For example, the clips about medical procedures may be used for teaching purposes, but if the clip shows a patient and his/her employer views it, it may have a detrimental effect, in particular, if the patient has a serious mental disease. While such multimedia clips may be of significant research value, appropriate measures need to be taken to ensure that privacy is protected. The three key factors of privacy protection are: *information sensitivity*, *information receiver,* and *information use* (Adams & Sasse, 1999; Jiang & Landay, 2002). Privacy policies indicate who receives what level of private information for what use. A relevant issue is the inference problem (Thuraisingham, 2003). For instance, disease information may be inferred by collecting information on prescription data related to a patient. In particular, data mining can be used to infer facts that are not obvious to human analysis of data. In such cases, we need to protect private information that may be inferred.

A model for user privacy: In general, an application domain provides a privacy policy to protect users' private information that may have been maintained as historical profiles or credential data collected for authorization purposes (Ashley, Hada, Karjoth, & Schunter, 2002; Jiang 2002; Tavani 2001). User consent is needed for releasing such private information to other parties. In particular, privacy requirements may need to be expressed as content-based and association-based constraints (Thuraisingham, 2003). A key challenge is thus to develop a privacy model and an enforcement mechanism that will be driven by the domain rules and user-controlled privacy policies. Suitability of several existing approaches to privacy management has not been applied to multimedia environments. The *sticky policy* based approach proposed in Ashley, Hada, Karjoth, and Schunter (2002) uses privacy

Figure 4. Privacy protection component



statements that *stick* to each piece of private information so that the privacy statements can be checked before the information is accessed. Another relevant approach uses privacy tags and defines an information space boundary (*physical*, *social*, or *activity-based*) as a contextual trigger to enforce permissions defined by the owners of that space (Jiang & Landay, 2002). To address flexible privacy requirements in multimedia applications, these approaches may need to be combined, and the use of parameterized roles can provide significant benefits (Kobsa & Schrek, 2003). One key issue is when a user specifies a privacy policy, the document itself becomes a protected object and may conflict with the overall privacy policy of the domain.

Privacy-aware information classification: To provide a fine-grained control over the release of private information, a mechanism is needed to provide the interaction between the data mining component that generates security clusters and the privacy protection module. Assume that a personal privacy policy states that "*None other than the concerned doctors should view any information that may reveal my disease*". Here, "*any information that may reveal my disease*" may simply be text containing a prescription drug name, the disease name, or a video sequence showing the patient's physical features and indicating signs of the disease; and the "*concerned doctors*" is an abstract concept referring to a user group with certain characteristics. Meta-information from the privacy-policy base can be used as the domain knowledge for supporting the information classifiers at the time of generating security classes and their attributes and feature sets. Furthermore, data mining can reveal new sensitive and private information and associations that may not be possible through human analysis. For such information, if privacy statements do not exist, new privacy policies may need to be generated. Hence, the development of mechanisms that allow efficient interactions between the privacy protection and data mining modules is needed to iteratively refine both privacy policies and information classification. The overall privacy protection component in the proposed research is depicted in Figure 4. The access control module interacts with the privacy mechanism to ensure that privacy requirements are met before users can access data. Several privacy protection models and approaches have been discussed (Adams & Sasse, 1999; Ashley, Hada, Karjoth, & Schunter, 2002; Ashley, Powers, & Schunter, 2002; Jiang & Landay, 2002; Kobsa & Schrek, 2003; Mont, Pearson, & Bramhall, 2003; Tavani & Moor, 2001).

Management of Secure Multi-Domain Multimedia Information Systems

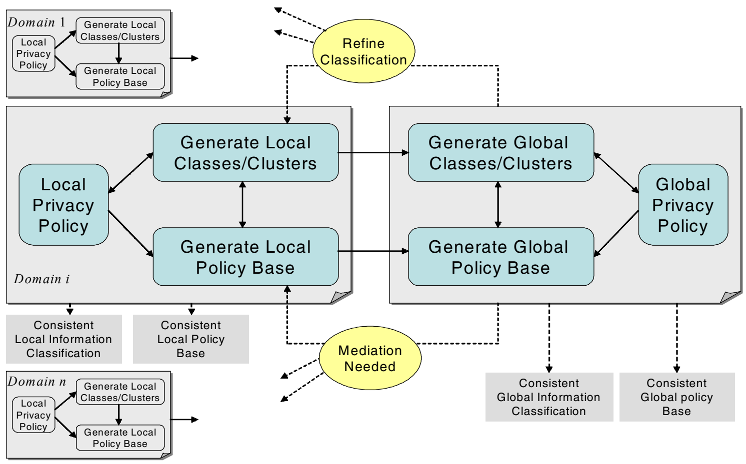
Heterogeneity is a uniquely complex challenge, as security and privacy policies and information classification methodologies may vary over all the domains giving rise to inconsistent inter-domain actions. The key challenges for developing secure distributed, multi-domain multimedia information systems are (1) to identify conflicts among security and privacy policies and (2) to maintain a consistent global information classification.

The overall infrastructure for a secure distributed multi-domain multimedia application environment must allow seamless and secure interoperation among the diverse access control policies employed by each domain. Such an infrastructure should be scalable, open, and extensible (Joshi et al., 2001) and should allow any domain with an arbitrary security policy and information classification to be a part of the overall environment. An XML-based policy specification and enforcement framework can provide a more practical solution for secure interoperation in multi-domain multimedia environments.

In a multi-domain environment, individual domain policies may follow different conventions and use varying attributes. Semantic heterogeneity may give rise to naming or structural conflicts, a problem that is commonly encountered during schema integration in a heterogeneous federated database system (Batini, Lenzerini, & Navathe, 1986). Naming conflicts occur due to the use of a single name to refer to different concepts. Known techniques from the integration of heterogeneous databases and ontology can be utilized to resolve naming conflicts (Batini et al., 1986). Conflicts can also arise because of the differences in the representation of similar constraints. In the CC-RBAC framework, the constraints can make the semantic heterogeneity issue even more difficult to address. It makes evolution management a significant problem.

Different application domains may use different data analysis and classification mechanisms to generate an equivalent set of concepts with different names or different concepts with similar names. At the global level, such semantic differences need to be properly resolved to ensure secure information access and privacy protection. In each application domain, there is an interaction between the information classification and policy generation modules as indicated in Figure 5. Once policies and information classification consistent to each other have been generated, the global level policy is generated. At the same time, a consistent global level classification also needs to be generated

Figure 5. The integration process



from local classifications. The global level policy and global level classification may be inconsistent if the local domains employ different classification schemes. To resolve such inconsistencies, local classifiers may need to be refined, which in turn affects the local policies. Hence, the global level policy generated earlier will now become inconsistent requiring further mediation of local policies. Such multi-dimensional horizontal and vertical interactions among the classification and policy generation modules at the two levels make the overall integration problem significantly complex. The challenge is to develop a framework for iteratively refining policies and information classes at the two levels. Local clusters/classes are extracted by data mining tools and are used as protection objects. Refinement of classifiers and mediation policies needs to be iterated until consistent classifications and policies are generated at each local domain as well as at the global level. The presence of dynamic and contextual roles can make the problem even more complex. Techniques are needed for detecting inconsistencies, automatically or semi-automatically, using attribute-based matching and ontologies. In particular, use of ontology to assist in semantic mapping between security policies of different domains is becoming a possibility because of tools such as RDF, DAML+OIL, and/or OWL. An ontology can be defined as "the specification of conceptualizations, used to help programs and humans share knowledge" (Hunter, 2003). Use of ontologies in the integration of security policies needs to be similarly explored. In particular, for securing a distributed multi-domain application, use of ontologies related to the domain knowledge as well as the security policies and mechanisms of the individual administrative domains will be crucial to capture the names used to indicate various security attributes and their intended semantics and relationships.

Limited work has been done in the area of secure interoperation in a multi-domain environment. In Shafiq, Joshi, Bertino, and Ghafoor (2004), a policy composition framework for integrating simple RBAC policies to provide secure interoperation has been proposed, where RBAC constraints are transformed into an integer programming problem to optimize secure interoperation. In Joshi et al. (2004), the XML-based RBAC framework has been extended to specify arbitrary multi-domain policies. Integration of multiple policies has been focused generally on MAC mechanisms for federated database systems. Thuraisingham and Ford (1995) discussed methodologies for constraint processing in a multi-level, secure, centralized, and distributed database management system. In Jonscher and Dittrich (1995), the global access layer is used to map global authorizations into local access rights for the individual systems. XML-based tools to manage RBAC policies in an enterprise have been discussed in Vuong, Smith, and Deng (2001).

Conclusions

In this chapter, we have presented a conceptual architectural framework for open, distributed multimedia information systems that aims at ensuring security and privacy of multimedia information and provides a mechanism for threat assessment. The proposed framework uses XML technologies to address the needs for multi-resolution representation of multi-modal multimedia data, its storage, exchange, and retrieval, as well as to develop access control and privacy specifications. We have discussed the relevance of a content and context based RBAC approach to capture both the access control and privacy requirements of multimedia applications. The framework addresses the need for a holistic solution to supporting flexible access control and privacy requirements on multimedia information. Such information may contain sensitive and private data and can be analyzed to reveal patterns and knowledge for possible threats to an enterprise. The framework provides security and privacy protection while facilitating real-time and long-term analysis to detect potential threats.

References

Adam, N.R., Atluri, V., & Bertino, E. (2002). A content-based authorization model for digital libraries. *IEEE Transactions on Knowledge and Data Engineering, 14*(2), 296-315.

Adams, A., & Sasse, M.A. (1999). Taming the wolf in sheep's clothing: Privacy in multimedia communications. *ACM Multimedia, 99*, 101-106.

Ahn, G., & Sandhu, R. (2000). Role-based authorization constraints specification. *ACM Transactions on Information and System Security, 3*(4), 207-226.

Aref, W.G., Catlin, A.C., Elmagarmid, A.K., Fan, J., Hammad, M.A, Ilyas, I., Marzouk, M., Prabhakar, S., & Zhu, X.Q. (2003). VDBMS: A testbed facility for research in video database benchmarking. *ACM Multimedia Systems Journal, Special Issue on Multimedia Document Management Systems*.

Ashley, P., Hada, S., Karjoth, G., & Schunter, M. (2002). E-P3P privacy policies and privacy authorization. *Proceedings of the ACM Workshop on Privacy in the Electronic Society*.

Ashley, P., Powers, C., & Schunter, M. (2002). From privacy promises to privacy management: A new approach for enforcing privacy throughout an enterprise. *Proceedings of the ACM New Security Paradigms Workshop '02*.

Atluri, V., Adam, N., Gomaa, A., & Adiwijaya, I. (2003). Self-manifestation of composite multimedia objects to satisfy security constraints. *ACM Symposium of Applied Computing,* 927-934.

Batini, C., Lenzerini, M., & Navathe, S.B. (1986). A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys, 18*(4), 323-364.

Bertino, E., Bonatti, P.A., & Ferrari, E. (2001). TRBAC: A temporal role-based access control model. *ACM Transactions on Information and System Security, 4*(3), 191-233.

Bertino, E., Castano, S., & Ferrari, E. (2001). Securing XML documents with Author-X. *IEEE Internet Computing,* 21-31.

Bertino, E., Castano, S., Ferrari, E., & Mesiti, M. (1999). Controlled access and dissemination of XML documents. *Proceedings of the 2nd International Workshop on Web Information and Data Management* (pp. 22-27).

Bertino, E., Hammad, M.A., Aref, W.B., & Elmagarmid, A.K. (2000). An access control model for video database systems. *CIKM,* 336-343.

Bhatti, R., Joshi, J.B.D., Bertino, E., & Ghafoor, A. (2003). XML-based RBAC policy specification for secure Web-services. *IEEE Computer*.

Burnett, I., De Walle, R.V., Hill, K., Bormans, J., & Pereira, F. (2003). MPEG-21 goals and achievements. *IEEE Multimedia*.

Chen, S.-C., Shyu, M.-L., Zhang, C., Luo, L., & Chen, M. (2003c). Detection of soccer goal shots using joint multimedia features and classification rules. *Proceedings of the 4th International Workshop on Multimedia Data Mining (MDM/KDD2003),* in conjunction with the *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 36-44).

Chen, S.-C., Shyu, M.-L., Chen, M., & Zhang, C. (2004a). A decision tree-based multimodal data mining framework for soccer goal detection. *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2004)*.

Covington, M.J., Long, W., Srinivasan, S., Dey, A.K., Ahmad, M., & Abowd, G.D. (2001). Securing context-aware applications using environmental roles. *Proceedings of the 6th ACM Symposium on Access Control Models and Technologies* (pp. 10-20).

Dagtas, S., & Abdel-Mottaleb, M. (2001). Extraction of TV highlights using multimedia features. *Proceedings of the IEEE International Workshop on Multimedia Signal*.

Damiani, E., di Vimercati, S.D.C., Paraboschi, S., & Samarati, P. (2002). A fine-grained access control system for XML documents. *ACM Transactions on Information and System Security, 5*(2), 169-202.

Dy, J.G., & Brodley, C.E. (2000). Feature subset selection and order identification for unsupervised learning. *Proceedings of the 17th International Conference on Machine Learning* (pp. 247-254).

Ferraiolo, D., Sandhu, R., Gavrila, S., Kuhn, R., & Chandramouli, R. (2001). The NIST model for role-based access control: Towards a unified standard. *ACM Transactions on Information and System Security, 4*(3).

Georgiadis, C.K., Mavridis, I., Pangalos, G., & Thomas, R.K. (2001). Flexible team-based access control using contexts. *Proceedings of the 6th ACM Symposium on Access Control Models and Technologies* (pp. 21-27).

Giacinto, G., Roli, F., & Didaci, L. (2003). Fusion of multiple classifiers for intrusion detection in computer networks. *Pattern Recognition Letters, 24*, 1795-1803.

Giuri, L., & Iglio, P. (1997). Role templates for content-based access control. *Proceedings of the 2nd ACM Workshop on RBAC* (pp. 153-159).

Hammad, M.A, Franklin, M., Aref, W.G., & Elmagarmid, A.K. (2003). Scheduling for shared window joins over data streams. *Proceedings of the International Conference on Very Large Data Bases (VLDB)* (pp. 297-308).

Hunter, J. (2003). Enhancing the semantic interoperability of multimedia through a core ontology. *IEEE Transactions on Circuit and Systems for Video Technology, 13*(1).

Jiang, X, & Landay, J.A. (2002). Modeling privacy control in context-aware systems. *IEEE Pervasive Computing, 1*(3), 59-63.

Jonscher, D., & Dittrich, K.R. (1995). Argos: A configurable access control system for interoperable environments. *Proceedings of the IFIP WG 11.3 9th Annual Working Conference on Database Security* (pp. 39-63).

Joshi, J.B.D., Bertino, E., Latif, U., & Ghafoor, A. (2005). Generalized temporal role based access control model. *IEEE Transactions on Knowledge and Data Engineering, 17*(1), 4-23.

Joshi, J.B.D., Bhatti, R., Bertino, E., & Ghafoor, A. (2004, November-December). An access control language for multidomain environments. *IEEE Internet Computing,* 40-50.

Joshi, J.B.D., Ghafoor, A., Aref, W., & Spafford, E.H. (2001). Digital government security infrastructure design challenges. *IEEE Computer, 34*(2), 66-72.

Joshi, J.B.D., Li, K., Fahmi, H., Shafiq, B., & Ghafoor, A. (2002). A model for secure multimedia document database system in a distributed environment. *IEEE Transactions on Multimedia, Special Issue on Multimedia Databases, 4*(2), 215-234.

Kobsa, A., & Schreck, J. (2003). Privacy through pseudonymity in user-adaptive systems. *ACM Transactions on Internet Technology, 3*(2).

Krzysztofowicz, R., & Long, D. (1990). Fusion of detection probabilities and comparison of multisensor systems. *IEEE Transactions on Systems, Man, & Cybernetics, 20*(3), 665-677.

Little, T.D.C., & Ghafoor, A. (1990). Synchronization and storage models for multimedia objects. *IEEE Journal of Selected Areas in Communications, Special Issue on Multimedia Communication, 8*(3), 413-427.

McGuinness, D.L., Fikes, R., Hendler, J., & Stein, L.A. (2002). DAML+OIL: An ontology language for the Semantic Web. *IEEE Intelligent Systems*.

Moni, S., & Kashyap, R.L. (1995). A multiresolution representation scheme for multimedia databases. *Multimedia Systems, 3*, 228-237.

Mont, M.C., Pearson, S., & Bramhall, P. (2003). Towards accountable management of identity and privacy: Sticky policies and enforceable tracing services. *Proceedings of the 14th International Workshop on Database and Expert Systems Applications* (pp. 377-382).

Qin, L., & Atluri, V. (2003). Concept-level access control for Semantic Web. *Proceedings of the ACM Workshop on XML Security*.

Sandhu, R., Coyne, E.J., Feinstein, H.L., & Youman, C.E. (1996). Role-based access control models. *IEEE Computer, 29*(2), 38-47.

Sandhu, R., Ferraiolo, D., & Kuhn, R. (2000). The NIST model for role-based access control: Toward a unified standard. *Proceedings of the 5th ACM Workshop Role-Based Access Control* (pp. 47-63).

Shafiq, B., Joshi, J.B.D., Bertino, E., & Ghafoor, A. (2004). Secure interoperation in a multi-domain environment employing RBAC policies. *IEEE Transactions on Knowledge and Data Engineering*.

Shyu, M.-L., Chen, S.-C., Chen, M., Zhang, C., & Sarinnapakorn, K. (2003). Image database retrieval utilizing affinity relationships. *Proceedings of the 1st ACM International Workshop on Multimedia Databases (ACM MMDB'03)* (pp. 78-85).

Tavani, H.T., & Moor, J.H. (2001). Privacy protection, control of information, and privacy-enhancing technologies. *ACM SIGCAS Computers and Society, 31*(1).

Thuraisingham, B. (2003). *Web data mining technologies and their applications to business intelligence and counter-terrorism.* Boca Raton, FL: CRC Press.

Thuraisingham, B., Clifton, C., & Maurer, J. (2001). Real-time data mining of multimedia objects. *Proceedings of the 4th International Symposium on Object-Oriented Real-Time Distributed Computing* (pp. 360-365).

Thuraisingham, B., & Ford, W. (1995). Security constraints in a multilevel secure distributed database management system. *IEEE Transactions on Knowledge and Data Engineering, 7*(2), 274-293.

Vuong, N., Smith, G., & Deng, Y. (2001). Managing security policies in a distributed environment using eXtensible Markup Language (XML). *Proceedings of the 16th ACM Symposium on Applied Computing* (pp. 405-411).

Zhu, X.Q., Fan, J., Aref, W., Catlin, A.C., & Elmagarmid, A. (2003). Medical video mining for efficient database indexing, management and access. *Proceedings of the 19th International Conference on Data Engineering* (pp. 569-580).

Chapter XII

Framework for Secure Information Management in Critical Systems

Rajgopal Kannan, Louisiana State University, USA

S. Sitharama Iyengar, Louisiana State University, USA

A. Durresi, Louisiana State University, USA

Abstract

The techniques described in this chapter will develop an understanding of three critical areas in sensor network security, namely, data confidentiality, anonymity, and integrity along with associated security-performance tradeoffs. These results should contribute toward the design of a security framework for a common sensornet architecture and enable the flexible deployment and use of sensor networks for a large variety of applications, including homeland security.

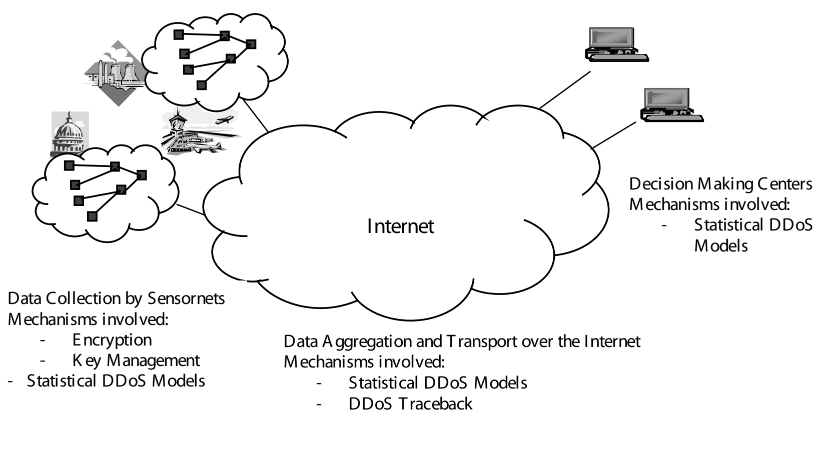
Introduction

The IT revolution has led to the development of sophisticated information systems that play a crucial role in the national economy. These information systems are not stand-alone; rather, they are fully distributed with connectivity through a mix of heterogeneous networks, for example, sensor networks, wireless and cellular networks, mobile and distributed systems, along with the Internet. Figure 1 illustrates a typical information system consisting of wireless sensor networks for gathering data, which is then transmitted to data processing sites over a wired backbone.

The changing trends in information management, diverse user information needs, and the widespread use of the Internet coupled with significant advances in hacking technology has made it crucial to shield information systems by developing robust trust and security schemes. Our information systems are characterized by the vast amounts of data generated continuously by their components, especially the large and easily deployable networks of small autonomous devices, and transported toward processing and dissemination centers. This whole process is vulnerable to serious security threats at various levels. Consequently, a major technical challenge is to design security solutions for managing this data over such heterogeneous environments.

Current research efforts are focused on solving the problem of data security in large information systems in an ad hoc or patchy manner. Such efforts do not

Figure 1. Information security mechanisms



account for the complex interaction between system components and leaves the system vulnerable to coordinated attacks. Due to the vital role of these information systems in the national economy, it is necessary to develop an integrated approach to system security that accounts for the joint vulnerabilities of the various components and protects the system against coordinated attacks. In this chapter, we discuss the development of such an integrated security framework by identifying and building locally optimal security mechanisms at various system layers and evaluating the interdependence between security mechanisms at different layers. Since sensor networks form an important data-gathering component of critical information systems, we describe our approach by examining the problem of security-driven sensor information management.

Security-Driven Sensor Information Management

Sensor networks (sensornets) are emerging as a critically important disruptive technology supported by recent technological advances in low power wireless communications along with silicon integration of various functionalities such as sensing, communications, intelligence, and actuations (Akyldiz, Su, Sankarasubramaniam, & Cayirci, 2002; Estrin, Girod, Pottie, & Srivastava, 2001; Estrin & Govindan, 1999). Their ability to acquire spatio-temporally dense data in hazardous and unstructured environments makes them attractive for a wide variety of applications (Estrin et al., 2001). Current research efforts have produced vertically integrated solutions tailored to specific applications such as environmental and seismic monitoring and battlefield scenarios like target detection and tracking, which is typical for an emerging technology. However, the tremendous range of potential sensornet applications implies that sensor systems must now evolve from an emerging technology to a dominant one, characterized by very large and scalable networks of interacting sensor systems and flexible, simple plug and play deployment. The eventual research goal is to enable sensor networks to become a commodity application.

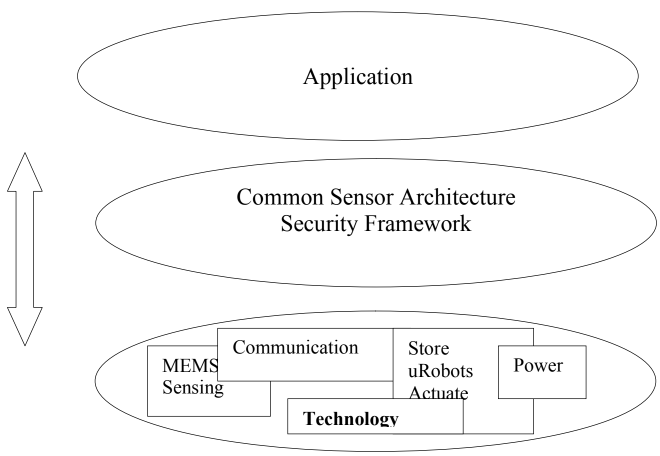
To achieve these goals and the full potential of sensor networks, a new network architecture is needed. The role of network architecture in this case is crucial because of the paradigm shift in computing brought about by sensor networks; the functionalities are not created from single nodes anymore but from the close collaboration of networked nodes, which make the network model (Estrin et al., 1999) more valid than ever before. The new network architecture, as shown in Figure 1, should be flexible, scalable, common for many applications, and extendable without the need to redesign the whole network. Other successful examples of such common, reusable network architectures are those of the Power Grid, Internet, and Ethernet.

Security is a crucial part of such a common sensornet architecture. Due to their nature and operational resource constraints, sensor networks are vulnerable to various types of attacks (Karlof & Wagner, 2003; Slijepcevic et al., 2002). While designing the new network architecture for future sensor networks, the research community has a unique chance to integrate security and privacy since the beginning is a fundamental part of the architecture. As shown by the Internet example, security cannot be implemented properly as patches to existing network architecture; rather, security mechanisms must be developed as part of an integral security framework.

In this chapter, we discuss the problem of developing such a security framework from two perspectives.

- The development of mechanisms for solving important sensor network security problems, namely, data confidentiality, integrity, and privacy. In

Figure 2. A common sensornet architecture



particular, we describe solutions for the problems of energy- and memory-efficient data encryption, algorithms for secure sensor communication using shared predistributed keys, mechanisms to ensure anonymous sensor-to sensor and sensor-to-sink data communication, and to maintain data integrity through secure data aggregation and Byzantine behavior detection in the presence of malicious nodes injecting false data.

- The proposed security framework must provide flexible tradeoffs between desired application-specific security levels and network resource consumption and combat a wide range of attacks. We specifically design these mechanisms to be interdependent, complementarily synergistic, and flexibly utilizable in appropriate combinations and develop algorithmic tools (such as security-performance metrics) for evaluating these interdependent and complementary properties.

Security Issues in Sensor Networks

Wireless sensor networks are often deployed in hostile or dangerous environments where nodes are not physically protected. This, along with the broadcast nature of the transmission medium, makes them extraordinarily vulnerable to a wide range of security threats. While security issues in ad hoc networks (Stajano & Anderson, 1999) are in many cases similar to those in sensor networks, ad hoc network defense mechanisms are typically not directly applicable to sensor networks. For example, some ad hoc network security mechanisms for authentication and secure routing are based on public key cryptography (Hubaux, Buttyan, & Capkun, 2001) which is too expensive for sensor nodes. Similarly, security solutions for ad hoc networks based on symmetric key cryptography have been proposed (Hu, Johnson, & Perrig, 2002). They are too expensive in terms of node state overhead and are designed to find and establish routes between any pair of nodes—a mode of communication not prevalent in sensor networks.

In this chapter, we develop a security framework for a common sensornet architecture by designing interdependent, complementary, and flexible security mechanisms for critical sensornet security threats. We first outline some of the more significant threats and the associated defense mechanisms.

- **Ensuring Data Integrity:** Nodes may be captured and reprogrammed, and their identities may be altered by adversaries. Once in control of a few nodes inside the network, the adversary can launch a variety of attacks, for example, injection of falsified sensing data and malicious code, thereby misleading legitimate users as well as consuming scarce energy sources. For example, compromised nodes under a geographic routing regime may advertise false location information, thereby becoming black holes. Thus, security mechanisms that ensure data integrity, along with mechanisms for ensuring valid sensor-to-sensor data/code transfer, are a crucial research area.
- **Protecting Data Content:** Data must be encrypted using energy-efficient techniques to prevent adversaries from accessing application-specific message content.
- **Guaranteeing Message Anonymity:** Though messages are encrypted, adversaries can use sophisticated traffic analysis to correlate data flow patterns with critical event locations within the sensornet. Mechanisms for anonymizing locations of sources and sinks for sensed data are therefore a necessary part of our security framework.

There are several recent research efforts exploring different aspects of sensornet security, for example, key management (Chan, Perrig, & Song, 2003; Du, Deng, Han, & Varshney, 2003; Eschenaur & Gligor, 2002), secure multicast communication, authentication and anonymous routing (Kong, n.d.). Among the original sensornet security solutions, SPINS (Perrig et al., 2001) presents two building block security mechanisms for use in sensor networks, SNEP and μ-TESLA. SNEP provides confidentiality and authentication between nodes and the sink, and μ-TESLA provides authenticated broadcast.

Note that sensor networks are vulnerable to a vast number of security threats with variable application-specific attack mechanisms and variable impact on the network. Therefore, while specific security solutions are useful, they do not defend against a variety of attacks and are not designed to flexibly exploit the complex interaction between various system components (such as current resource constraints vs. application-specific security needs, etc.). Indeed, such solutions might leave the system vulnerable to coordinated attacks. Therefore, building future sensornet common architectures will require a large variety of security mechanisms corresponding to unique sensornet vulnerabilities in the areas of data confidentiality, integrity, and privacy. More importantly, these security mechanisms should have the following properties:

- **Interdependency:** Strong security mechanisms at one level can be coupled with relatively weaker mechanisms at another less important level. For example, two major aspects of data confidentiality are the encryption of sensitive data and its authenticated exchange between sensors. Due to energy constraints, encrypted communication between sensors is typically via shared symmetric keys, an implicit form of authentication. There is an obvious correlation between the strength of data encryption schemes and the nature of the shared keying mechanism between sensors. Investigating these tradeoffs will enable the design of encryption schemes optimally suited to the accompanying keying mechanisms. Later we describe solutions that ensure a high degree of secure network connectivity and are resilient to node/key capture.
- **Complementarity:** Complementary security solutions with synergistic combinations are needed to defend against variants of the same type of attack. Attacks with the same objective can take varying forms and have different frequencies/likelihoods based on the nature of adversaries, such as energy and other costs to the adversary, his/her intelligence level, and so forth. For example, consider the problem of data integrity in which sensornet data can be compromised by an adversary in several ways. In one instance, transmitted data can be maliciously corrupted en route by captured sensor nodes. Alternately, under a more intelligent adversarial model, sensor nodes can deliberately inject falsely sensed information. We develop complementary and synergistic solutions based on statistical data validation for aggregated sensornet data at the application layer which detects falsified data and potentially defective/insecure sensors along with lightweight network layer solutions which protect against corruption of transmitted information enroute to the sink.
- **Flexibility:** Combinations of security solutions should be flexibly utilizable based on the current needs of the application, the current resource constraints, and functional support available from the network and the likelihood of different categories of attack. For example, different sensornet applications require different degrees of data anonymity yet should be supported within the common sensornet architectural framework. We develop solutions that provide flexible levels of data anonymity (subject to application-specific needs) traded off with energy consumption at

sensors. Similarly, our data integrity solutions tradeoff the likelihood of specific attacks with the energy cost of data validation solutions while our proposed encryption mechanisms are designed to tradeoff encryption power with memory capacity (in terms of key storage at sensors).

Technical Challenges

Due to their large scale, autonomous operation, and massive amounts of data flow over insecure wireless channels, sensor networks are significantly vulnerable to data attacks by adversaries. Future common sensor network architectures will therefore require a security framework comprising security mechanisms that address the problems of *data confidentiality, integrity,* and *anonymity*. These mechanisms must also be designed to be interdependent and complementary and provide application-specific security performance flexibility. In this chapter, we describe novel and resource-efficient solutions as part of this framework.

- **Confidentiality-Secure Key Management:** Given the resource constraints on sensors, the encryption process for communication between neighboring sensors is via symmetric keys. Since online (post-deployment) key distribution/exchange is energy inefficient, key predistribution, that is, preinstalling a limited number of keys in sensor nodes prior to actual deployment is an attractive solution. However, capture of a node might result in full knowledge of the keys stored at that node, thereby compromising all communication between other nodes in the network based on known keys. Therefore, a key technical challenge is to develop novel key predistribution mechanisms which balance the requirements of good network connectivity, resilience to enemy attack, memory-efficiency, and scalability.
- **Anonymity:** Guaranteeing anonymity is an important security feature in many types of sensor networks that operate in hostile environments. Using sophisticated traffic and data flow analysis, attackers can infer the location of active event areas, sources, and sinks and thereby disrupt the functionalities of the network. A major sensornet security issue is to anonymize sources and sinks in sensor networks despite the inability to

hide wireless transmissions. Here, we describe anonymization protocols in which location information is diluted among many nodes, thereby making it difficult for attackers to locate the origin and destination of data. One major advantage of our protocol is that it enables the user to tradeoff overhead with anonymity, making the solution flexible to application requirements.

- **Data Integrity:** Sensor networks are expected to consist of hundreds to thousands of nodes dispersed in hostile environments. It is clearly impractical to monitor and protect each individual node from physical or logical attack. An enemy can easily alter existing data or inject spurious data in the sensornet by capturing or inserting malicious nodes into the network. A key technical challenge is to detect such activity by distinguishing fake/altered data from the correct ones and identifying the malicious nodes. In data-centric sensor networks, data is typically aggregated for energy efficiency. Since sensor networks are highly unstructured, both routing and aggregation of data occurs in an ad hoc manner depending on current resource distributions and current (localized) sensing activity. It is therefore extremely difficult to identify vulnerable nodes/network zones *a priori*. Therefore, there is a need to develop a broad spectrum of dynamic defense mechanisms for detecting such malicious behavior. In this chapter, we describe complementary solutions for ensuring application-level data integrity using statistical techniques for validating aggregated sensornet data along with lightweight network layer solutions for detecting malicious data corruption while forwarding.

Key Predistribution for Secure Sensornet Communication

Key Predistribution Problem

Sensor nodes are usually deployed in an ad hoc manner into arbitrary topologies before self-organizing to form a sensor network (Estrin et al., 1999). Establishing a secure communication infrastructure among sensor nodes is a challenging problem, known as the bootstrapping problem (Chan et al., 2003). Traditional methods of key exchange and key distribution protocols based on trusted third-party mechanisms are infeasible, since it is impossible to predetermine the neighborhood of any node. Moreover, given the limited memory capacity at sensor nodes, key distribution schemes must ensure good network connectivity, even with limited number of keys per node. One attractive possibility for ensuring secure communication between adjacent sensor nodes is key predistribution.

There have been several recent works on key predistribution (Chan et al., 2003; Du et al., 2003; Eschenaur et al., 2002; Liu & Ning, 2003; Zhu, Setia, & Jajodia, 2003). The pioneering paper in Karlof and Wagner (2003) proposes a simple, scalable probabilistic key predistribution scheme in which a certain number of keys are drawn at random from a (large) key pool and distributed to sensor nodes prior to their deployment. Post-deployment, adjacent nodes participate in shared key discovery, thereby creating a logical graph. This is followed by the establishment of secure paths between nodes using secure links in the logical graph. In Chan et al. (2003), the authors have presented new mechanisms for key establishment using the random key predistribution scheme of Eschenaur et al. (2002) as a basis. Their q-composite scheme requires that two adjacent communicating nodes have at least q keys in common. This scheme provides high resiliency against small scale enemy attack.

Proposed Approach: Two-Phase Key Predistribution mechanism

We now describe a novel key predistribution scheme labeled 2-Phase in which sensor nodes are preloaded with a combination of *randomly derived* and *inherited keys*. The 2-Phase scheme is based on Kalidindi et al. (2004) and Kannan, Ray, and Durresi (2004) where we show superior network resilience to node capture and superior security-performance tradeoffs compared to random key predistribution. Two-Phase is motivated by the following key observations:

- From the connectivity point of view, the probability of having a common key between two nodes decreases as the key pool size increases under the random predistribution scheme. We observe that the (probabilistic) connectivity of the logical graph can be increased if we can ensure that each node *deterministically* shares some of its keys with some nodes, as in our subvector scheme (Kalidindi et al., 2004).

- We hypothesize that it is better from the security point of view to predistribute keys in a less-random fashion such that whenever a node shares a key with another node, it should be likely to share a larger number of keys with this node. If so, the resulting network should consist of high-composite links. Note that q-composite schemes are more secure with increasing q. If the adversary has obtained X keys (through the capture of one or more sensor nodes), the probability of determining the exact q subset of X that is used by a given communicating sensor pair decreases exponentially with increasing q.

We now describe the key steps in the proposed 2-Phase key distribution mechanism. Order the sensor nodes *a priori* in a logical queue and distribute keys in increasing order according to the following rules: The first node is assigned k keys drawn randomly from the key pool of size L ($K = \Theta(\sqrt{(L)})$) (offers good security-performance tradeoffs). For every succeeding sensor node *i, k* keys are distributed in two consecutive phases. First, node *i* receives a predetermined fraction f (1/k ≤ f < 1) of its k keys drawn randomly from the key space of node *i*–1. The remaining (1–f) fraction of k keys are then drawn randomly from the key pool of size L–k, *after excluding all k keys of node i–1 from L*.

The 2-Phase scheme is designed to be biased in favor of nodes sharing several keys with their immediate predecessors and successors, through direct inheritance as well as a random component. Intuitively, this key predistribution methodology should offer better secure connectivity in the logical graph by inducing the sharing of a larger number of keys between nodes, thereby enabling *q*-composite communication for larger values of *q*. More surprisingly, however, this methodolgy also provides enhanced security under node capture/eavesdropping by allowing for more exclusive key sharing between communicating nodes. The fraction *f* (called *inheritance ratio*) plays a significant part in the connectivity/security of the logical graph created after node deployment. Note that the random key predistribution scheme *is not* a special case of the 2-Phase scheme with *f* = 0, since we eliminate all k keys of the previous node, regardless of the value of *f*. The proposed scheme is also scalable since new sensor nodes can be assigned keys according to this rule at any time.

Security-Performance Tradeoffs

Security mechanisms directly impact system performance. Therefore, we need to develop a rigorous analytical framework for measuring the security-performance tradeoffs of arbitrary key distribution schemes. In particular, we will develop quantitative metrics that measure network resiliency to node/key capture, the number of available secure links, and the key (memory) requirement per node for a given level of connectivity.

Using standard probabilistic analysis in Kannan et al. (2004), we have proven several results that indicate the efficacy of these metrics in measuring security-performance tradeoffs. We compare network connectivity and security performance and show analytically and through simulations that the proposed 2-Phase scheme strongly favors highly secure large-composite key communication and is more resilient to node capture than the random scheme. It can be shown analytically that the invulnerability of an arbitrary q-composite communication link to any number of node captures is higher in our scheme. We derive analytical results for measuring the vulnerability of a q-composite link to single-node capture assuming adversaries who can use captured-key knowledge network-wide, as well as locally, and show that the 2-Phase scheme is more resilient. We summarize some of these results (without proofs) below:

Proposition 1: Two neighboring sensors are more likely to share at least one key under the 2-phase scheme as opposed to random key predistribution.

Proposition 2, Scalable Comparative Exclusivity: The invulnerability of a communication edge under any number of node captures when keys are distributed using the 2-Phase scheme with key replacement scheme is

$$IV^{2PWR} = IV^{Rand} \frac{(1-f)^4}{\left(1-f\frac{k}{L}\right)^{N-1}}$$

Thus, link invulnerability under 2PWR outperforms the random scheme as the size of the sensornet N scales upward.

Proposition 3: The vulnerability of a given communication link in an N-node sensor network with parameters k and L is lower if keys are predistributed using the 2-Phase scheme as compared to random key *predistribution,*

$$\frac{1}{k} \le f \prec x\frac{2-x}{1+2x}, x = k/L$$

Figure 3 presents some simulation results on the security-performance aspects of the proposed 2PWR scheme on a 1024 node sensornet arranged in clusters (Handy, Haase, & Timmerman, 2002) of varying sizes from 20-50 nodes. The 2-phase scheme outperforms the random one in many of the above security-performance metrics.

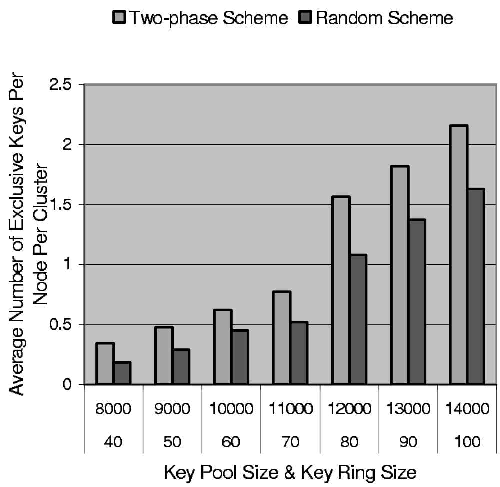
Anonymity in Sensor Networks

Guaranteeing anonymity is an important security feature in many types of sensor networks that operate in hostile environments. Using traffic analysis based on wireless tracking, it is possible for the attacker to infer the location of sources and destinations. The information gained can then be used to disrupt the functionalities of the network. We describe a novel approach for hiding the senders and receivers of messages.

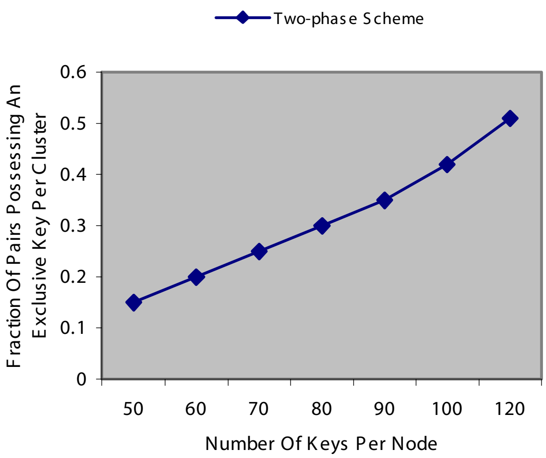
Related Work

Earlier works (Akyldiz et al., 2002; Gruteser & Grunwald, 2003) have presented protocols that enable each node to keep its location and its identifier hidden from other network nodes in a mobile ad hoc network. We consider a more general model in which the attacker (who may not be part of the network) has access to the entire network's traffic information and can use traffic analysis to infer who is talking to whom. In Kong, the author proposes a solution for untraceable routing. However, attackers may still be able to infer where data is generated by analyzing the chronology of transmissions. We propose mechanisms that are resilient to traffic analysis, that is, make it difficult for observers to learn identifying information about the source/sinks.

Figure 3. (a) Average number of exclusive keys per node pair; (b) Probability a node pair has an exclusive key

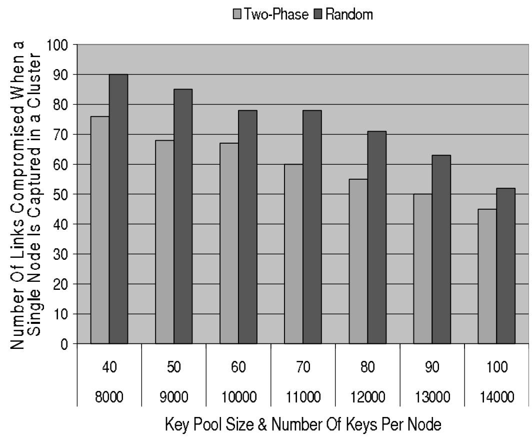


(a)

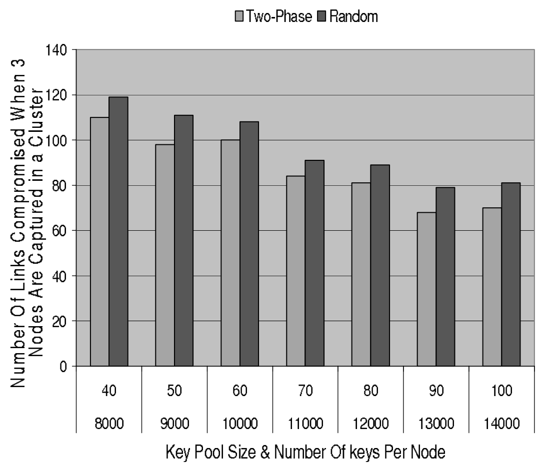


(b)

Figure 4. Average number of links compromised in a cluster when (a) one and (b) three nodes are captured



(a)



(b)

Proposed Approach: Anonymizing Sources and Sinks

We assume that the sensor network is clustered, and communication takes place via cluster heads (Handy et al., 2002). Cluster heads (called nodes in this section) employ rings (assumed to be created by the underlying sensornet routing protocol) to send messages to the sink and use a token passing access mechanism to insert a message on a ring. We assume that each node shares a symmetric key with its neighbor in the ring. When a node gets control of the token, it may transmit data in the message. Each message is of fixed length and contains the status of the token itself. The format of a message is as follows:

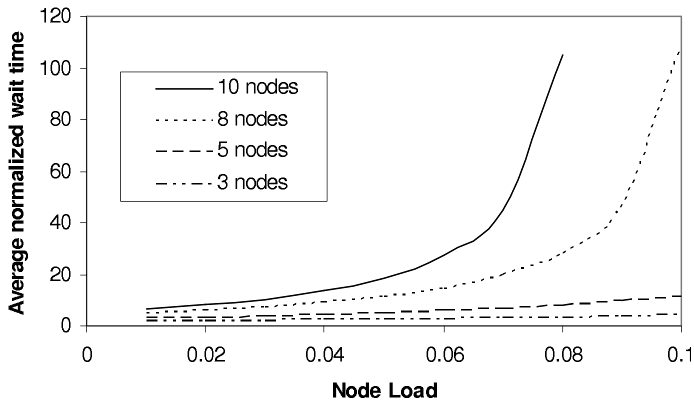
<E((Token||E(Message_Header, Ksd)||E(Message_Data, Ksd)), Ksi)>

where Ksi is the secret key shared between the source node s and upstream neighbor i. Ksd is the secret key shared between the source node s and the destination d. The token

includes the Status of token, which specifies if the token is free or occupied. If a token is free, a node can send data through that message, or else it cannot. The Message_Header includes source and destination addresses. Both Token and Message_Header use Redundancy predicates to check their validity. The Message_Data includes Data length, Data, and Padding. Data length specifies the length of the total data in the packet. This is crucial when the amount of data is not enough to fill the whole frame, in which case, the data field is padded with some random number to meet the fixed message length constraint.

Whenever a node i receives a message, it decrypts it using the shared key with its upstream node and verifies the redundancy predicate. Once the Redundancy predicate is fulfilled, the status of the token is checked. If the status is free and the node i has data to send to d, it will construct the message and set the status field to occupied. If the status of token is occupied, the node checks if the message is addressed to itself, in which case, it will make a copy and release the message upstream. Once the message returns to the source, the source repeats the procedure as long as it has data to send. When it has no more data to send, it sets the status field of the token to free and sends the padded message on the ring.

Figure 5. Waiting time vs. number of nodes on the ring



Results

The size of the rings, the number of nodes that generate data on a ring, and the number of tokens on a ring offer a large range of tradeoffs between security and communication performance. Figure 5 shows the waiting time for a message (normalized to transmission

time of the message) as a function of normalized latency of the ring, which depends on the number of nodes and normalized node load.

It is clear that more nodes on the ring create more communication overhead and increase waiting time but increase anonymity. We measure anonymity using a new metric, Exposure Degree (ED), defined as the ratio between the number of nodes generating data on the ring to the total number of nodes on the ring. Lower ED is achieved by having more passive nodes on the ring thereby reducing the chances of the attacker to identify the data sources.

Extensions

In order to enable several nodes to communicate simultaneously, we propose to introduce multiple frames in a ring with each frame having its own token. This improves the efficiency of the network and also enables multiple simultaneous communications. Also, each node in the ring is assigned a priority level. This allows nodes with high priority to use the network more frequently and enable more important traffic to be transmitted immediately.

In general, it is not preferable to have a lot of data source nodes transmitting on a ring at the same time. One reason is the token wait time: nodes might have to wait for a long time to get a hold of a token, resulting in large delays. On the other hand, the presence of more nodes increases the ED. In the worst case scenario, every node is sending a message to the base station. In this case, the attacker's assumption that every node is transmitting is perfectly valid. To prevent a ring from having a lot of transmitting nodes, we would like to reconfigure the rings dynamically. This can be done by the base station which knows which cluster heads are transmitting.

Data Integrity: Application-Layer Security with Data Aggregation

Data aggregation is one of the most important data-centric energy-conserving mechanisms within a sensor network (Akyldiz et al., 2002; Krishnamachari, Estrin, & Wicker, 2002). A set of special nodes, called *aggregators* (desirably less energy-constrained), perform in-network data aggregation and get rid of redundant data, thereby enabling energy-efficient information flow. The task of data aggregation becomes challenging when sensors and aggregators are deployed in a hostile environment and become subject to physical tampering. The foremost requirement of such a data processing mechanism is resilience to enemy attacks. We address the problem of enabling secure information aggregation at the application layer, given that a fraction of the aggregators and ordinary sensors might be compromised.

An adversary can perform a variety of node compromising attacks such as DoS, spoofing, Black Hole, and so on (Karlof & Wagner, 2003; Slijepcevic et al., 2002; Stinson, 2002) to affect data availability in a sensornet. Here we consider a different kind of attack on data integrity, called *stealthy attack* (Przydatek, Song, & Perrig, 2003), where the attacker aims to make the user accept false information, significantly different from the true data. We consider the following setup for this problem. A large number of sensors are deployed in an area distant from a *home server* which would like to get information about that area covered by the sensors. However, sensors are typically simple, lowpowered devices which can communicate within a small range of their location and are thus unable to send information directly to the home server. In such a situation, one or more resource-enhanced base stations can be used as intermediaries between the home server and ordinary sensors. We assume that there are several such base stations in a sensornet who collect information from all the sensors, process data, and forward to the home server.

Proposed Approach

We will solve the problem of secure information processing in this setup in two different stages.

1. In the first stage, an aggregator collects raw information from a set of sensors. A fraction of these sensors can be corrupt/faulty and thus send wrong values. *The task of the aggregator is to aggregate the raw data in such a way that the aggregated data is correct with high probability.*
2. In the second stage, an aggregator reports the aggregated data to a user/ home server. *The task of the base station is to detect with high probability whether this aggregator is corrupt or not.*

As a solution to the first part of the problem, we propose a novel probabilistic data accuracy model based on the following assumptions:

- **Spatial Locality:** Data collected by proximate sensors, whether good or corrupt/faulty, is likely to be differentially correlated by sensor distance. That is, the difference between two data values reported by two sensors is proportional to the distance between these two sensors if they are in the same neighborhood.
- **Statistical Fault/Corruption Model:** Each sensor has a probability q of being corrupt or faulty.

Therefore, given the data value from a node, we can measure the accuracy of information sent by all its neighbors. We outline our formal approach below (Ray, Kannan, & Durresi, 2004):

Let S={$s_1, s_2, ..., s_n$} be the set of sensors in the sensor network. Let si and sj be neighbors at a distance d(s_i, s_j). The difference between the value of information sent by these two nodes is denoted by D(s_i, s_j). We assume that D(s_i, s_j) varies exponentially with distance, that is, $k^{d(s_i,s_j)}$, where k is an empirical parameter. (Note that for certain sensornet data, the relationship might be linear, in which case, the model can be easily modified). The accuracy of information sent by the node is defined as follows:

$$Acc(s_i \mid s_j \text{is uncorrupt}) = \begin{cases} 1 & \text{if } D(s_i, s_j) \le k^{d(s_i, sj)} - \varepsilon \\ e^{-ar} & \text{if } k^{d(s_i,s_j)} - \varepsilon \le k^{d(s_i,s_j)} + \varepsilon \\ 0 & \text{if } D(s_i, s_j) \ge k^{d(s_i,s_j)} + \varepsilon \end{cases} \quad \textbf{(1)}$$

Similar formulations can be derived for Bayesian accuracy given compromised sensors. On receiving raw information from all the sensors, an aggregator will construct a table of relative accuracies of all the sensors and use this to compute the aggregated information value as the (appropriately) weighted average of information values sent by all the sensors.

The second part of the problem has been addressed in Przydatek et al. (2003). However, the main solution in Przydatek et al. (2003) was provided for the single aggregator setup, which is vulnerable to single node capture/compromise. We must therefore derive solutions for the secure aggregation problem under a more generalized and secure setup where multiple aggregators act as intermediaries between a group of sensors and a home server. In this setup, each aggregator receives raw data from all the sensors in the group and sends the aggregated data to the server. If a majority of aggregators are known to be incorrupt, then the server will simply accept the data value with the highest frequency. *The challenge for the server is to accept the correct value with high probability when there is no information about how many and which aggregators are corrupt*. A natural solution to this problem is to test the hypothesis that the highest frequency value is the value of the actual parameter (weighted mean) of the population consisting of all sensor values/readings. We sketch the outline of our proposed algorithm as follows:

- The server selects a fraction f_1 of aggregators, which sends the majority data value, and a fraction f_2 of remaining aggregators.

- It requests each of those aggregators to send a random sample of size n of the raw data that they collected from all the sensors.
- It performs the testing of the hypothesis that the majority weighted average is the actual weighted average of the whole population.
- It accepts of rejects the hypothesis with a prefixed probability.

Note that any testing of hypothesis can make errors of two types: rejecting a true hypothesis and accepting a false hypothesis. The fractions f_1 and f_2 and the sample size n should be determined in such a way that the probability of having these two errors are minimized as much as possible.

Conclusions

Sensor networks are emerging as a critically important disruptive technology that are ubiquitously deployable and support a large variety of applications, ranging from security monitoring in airports to the detection of hazardous biological agents. Current research efforts have produced vertically integrated solutions tailored to specific applications, which is typical for an emerging technology. To achieve their full potential and transform sensor networks into a commodity application, a new common network architecture is needed, and security should be an integrated part of such an architecture.

The techniques described in this chapter will develop an understanding of three critical areas in sensor network security, namely, data confidentiality, anonymity, and integrity along with associated security-performance tradeoffs. These results should contribute toward the design of a security framework for a common sensornet architecture and enable the flexible deployment and use of sensor networks for a large variety of applications, including homeland security.

Acknowledgments

This work was supported in part by NSF grants IIS-0312632 and IIS-0329738.

References

Akyldiz, I.F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002, March). Wireless sensor networks: A survey. *Computer Networks, 38*(4), 393-422.

Capkun, S., Hubaux, J.P., & Jakobsson, M. (2004, January). *Secure and privacy-preserving communication in hybrid ad hoc networks* (EPFL-IC Tech. Rep. No. C/2004/10).

Chan, H., Perrig, A., & Song, D. (2003). Random key pre-distribution schemes for sensor networks. *IEEE Symposium on Security and Privacy*.

Du, W., Deng, J., Han, Y.S., & Varshney, P.K. (2003). A pairwise key pre-distribution scheme for wireless sensor networks. *Proceedings of the 10th ACM Conference on Computer and Communication Security*, October 27-30, Washington, DC.

Eschenaur, L., & Gligor, V.D. (2002). A key-management scheme for distributed sensor networks. *Proceedings of the 9th ACM Conference on Computer and Communication Security,* November (pp. 41-47).

Estrin, D., & Govindan, R. (1999, August). Next century challenges: Scalable coordination in sensor networks. *Proceedings of the ACM/IEEE Conference Mobicom'99,* August (pp. 263-270).

Estrin, D., Girod, L., Pottie, G., & Srivastava, M. (2001). Instrumenting the world with wireless sensor networks. *Proceedings of the 10th International Conference on Acoustics, Speech and Signal Processing (ICASSP 2001)*, July.

Gruteser, M., & Grunwald, D. (2003, May). Anonymous usage of location-based services through spatial and temporal cloaking. *Proceedings of the 1st International Conference on Mobile Systems, Applications, and Services,* May.

Handy, M.J., Haase, M., & Timmermann, D. (2002). Low energy adaptive clustering hierarchy with deterministic cluster-head selection. *Proceedings of the 4th IEEE International Conference on Mobile and Wireless Communications Networks*.

Hubaux, J., Buttyan, L., & Capkun, S. (2001). The quest for security in mobile ad hoc networks. *Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC 2001)*.

Hu, Y.-C., Johnson, D.B., & Perrig, A. (2002). SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks. *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2002),* June (pp. 313-X).

Kalidindi, R., Parachuri, V., Kannan, R., Durresi, A., & Iyengar, S. (2004). Sub-quorum based key vector assignment: A key pre-distribution scheme for wireless sensor networks. *Proceedings of the International Conference on Wireless Networking (ICWN04)*, July, Las Vegas, Nevada.

Kannan, R., Ray, L., & Durresi, A. (2004). An efficient two-phase key predistribution scheme for wireless sensor networks. *Proceedings of the 1st European Workshop on Security in Sensor Networks*, August, Heidelberg, Germany. Retrieved May 7, 2005, from *http://arxiv.org/pdf/cs.NI/0405035*

Kannan, R., Ray, L., Iyengar, S.S., & Kalidindi, R. (2004). Max-min length-energy-constrained routing in wireless sensor networks. *Lecture Notes in Computer Science, 2920*, 234-249.

Karlof, C., & Wagner, D. (2003). Secure routing in sensor networks: Attacks and countermeasures. *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications,* May 11, Anchorage, Alaska.

Kong, J., Hong, X., & Gerla, M. (n.d.). *An anonymous on demand routing protocol with untraceable routes for mobile ad-hoc networks* (Tech. Rep. No. 030020). UCLA Computer Science Department.

Krishnamachari, B., Estrin, D., & Wicker, S. (2002). Modeling data-centric routing in wireless sensor networks. *Proceedings of the Wireless Sensor Network Applications,* June.

Liu, D., & Ning, P. (2003). Establishing pairwise keys in distributed sensor networks. *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03),* October, Washington, DC.

Perrig, A., Szewczyk, R., Wen, V., Culler, D., & Tygar, J.D. (2001). Spins: Security protocols for sensor networks. *Proceedings of the 7th International Conference on Mobile Computing and Networks,* July.

Przydatek, B., Song, D., & Perrig, A. (2003). SIA: Secure information aggregation in sensor networks. *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems* (pp. 255-265).

Ray, L., Kannan, R., & Durresi, A. (2004). Bayesian accuracy and hypothesis testing models for secure information aggregation in wireless sensor networks. *Proceedings of the ISNG 2004,* October, Las Vegas, Nevada.

Slijepcevic, S., Tsiatsis, V., Zimbeck, S., Srivastava, M.B., & Potkonjak, M. (2002). On communication security in wireless ad-hoc sensor networks. *Proceedings of the 11th IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE),* June (pp. 139-144).

Stajano, F., & Anderson, R.J. (1999). The resurrecting duckling: Security issues for ad-hoc wireless networks. *Proceedings of the 7th International Security Protocols Workshop* (pp. 172-194).

Stinson, D. (2004). *Cryptography: Theory and practice* (2nd ed.). DIMACS Series, Vol. 18. Boca Raton, FL: CRC Press.

Zhu, S., Setia, S., & Jajodia, S. (2003). LEAP: Efficient security mechanisms for large-scale distributed sensor networks. *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, October, Washington, DC.

Chapter XIII

Trustworthy Data Sharing in Collaborative Pervasive Computing Environments

Stephen S. Yau, Arizona State University, USA

Abstract

Collaborative Pervasive Computing Applications (COPCAs) can greatly improve the investigative capabilities and productivity of scientists and engineers. Users of COPCAs usually form groups to collaboratively perform their tasks using various computing devices, including desktop computers, pocket PCs, and/or smart phones, over Mobile Ad hoc Networks (MANET), LAN, and the Internet. These users usually share various types of data, including research ideas (documents), experimental and statistical data (numerical data, graphics, stream audio/video). A very important issue for sharing data in Collaborative Pervasive Computing Environments (COPCEs) is trustworthiness. To support trustworthy data sharing among groups of users of COPCAs, secure group communication, trustworthy

shared data discovery*,* flexible access control mechanisms, effective data replication*,* data quality assurance mechanisms*, and* intrusion detection mechanisms *are needed. In this chapter, the challenges, current state-of-the-art, and future research directions for trustworthy data sharing in COPCEs are presented. In particular, discussions will be focused on research issues and future research directions for* trustworthy shared data discovery *and* flexible access control *in service-based COPCAs.*

Introduction

Collaborative Pervasive Computing Applications (COPCAs), such as collaborative research and development environments, can greatly improve the investigative capabilities and productivity of scientists and engineers in many fields. Users of COPCAs usually form groups (or teams) to collaboratively perform their tasks. The collaborations are supported by the users' computing devices, such as desktop computers, pocket PCs, and/or smart phones, over various networks like Mobile Ad hoc Network (MANET), LAN, and the Internet. Users of COPCAs usually need to share various types of data, including research ideas (documents), experimental and statistical data (numerical data, graphics, stream audio/video). Data sharing among various groups of computing devices is one of the most important requirements for COPCAs because data sharing is required for efficient and effective group collaboration. For example, the status of each group member often needs to be shared for group collaboration, and all group members often need to share the same view of certain data when they collaborate on certain tasks.

During the past several years, research on mobile and pervasive data management has generated many useful results for managing and sharing data in pervasive computing environments. A significant trend in designing large-scale information systems in heterogeneous pervasive computing environments, consisting of multiple organizations, is utilizing Web services and emerging Semantic Web technology to improve interoperability and greatly enhance automated data composition/integration. However, most research focuses on how to improve the efficiency of discovering, accessing, distributing, and/or updating shared data. A critical issue, which has not attracted much attention, is trustworthiness of data sharing in COPCEs.

Data sharing in COPCEs has the following trustworthiness requirements: (a) flexible and adaptive access control to shared data, (b) secure and reliable shared data discovery, (c) secure and private data retrieval/delivery, (d) authentication of group member devices, (e) scalable and lightweight group key management, (f) detection of attacks and malicious users, (g) high availability of shared data, (h) shared data quality assurance, and (i) inference prevention. Development of effective mechanisms to satisfy these requirements is challenging due to the following reasons:

1. The mechanisms must be lightweight due to the severe resource constraints of pervasive computing devices.
2. The mechanisms must be highly and dynamically scalable due to the possible large variable number of pervasive computing devices in collaboration.
3. The mechanisms must be able to transparently support the device mobility due to high mobility of pervasive computing devices.
4. The mechanisms must be capable of performing runtime adaptation based on the situation; that is, they must be situation aware due to vulnerable and unstable wireless networks, and the pervasive ephemeral computing environments. We consider a *situation* as a set of past contexts of individual devices relevant to future device actions, and a context is any instantaneous, detectable, and relevant condition of the environment, device, or user (Yau, Wang, & Karim, 2002).

Substantial research has been done in the areas of secure group communication (Berket & Agarwal, 2003; Chiang & Huang, 2003; Lazos & Poovendran, 2003; Yau & Zhang, 2004), effective data replication (Guy et al., 1998; Ratner, Popek, & Reiher, 1999), and intrusion detection mechanisms (Kyasanur & Vaidya, 2002; Zhang, Lee, & Huang, 2003). Results from these research efforts can be applied to address the requirements (c) through (g) in COPCEs. Novel techniques for data quality assurance [Requirement (h)] and inference prevention [Requirement (i)] in trustworthy data sharing in COPCAs need to be developed to further improve the trustworthiness of shared data. Due to limited space, the research on these aspects will not be covered in this chapter.

In this chapter, we will focus on the current state-of-the-art research issues and future research directions for addressing requirements (a) and (b). In particular, our discussion will emphasize the Web service-based information systems mentioned earlier due to increasing popularity and deployment of service-based systems.

Current State-of-the-Art

As mentioned in the last section, our discussion will emphasize Web service-based information systems in pervasive computing environments. In these Web service-based information systems, shared data is published in the form of Web services and can be accessed by users through well-defined service interfaces. Hence, the trustworthiness of shared data in these information systems relies on the trustworthiness of services for data sharing. In this section, we will briefly review the state-of-the-art in the following three aspects related to the trustworthiness of services for data sharing: service specification languages, service discovery/matchmaking, and access control. Although much research has been done on service specification languages (Andrews et al., 2003; Atkinson et al., 2002; Della-Libera et al., 2002; OASIS 2003b, 2004; W3C, 2001, 2004b), service discovery (Balazinska, Balakrishnan, & Karger, 2002; Chen & Kotz, 2003; Czerwinski et al., 1999; Paolucci et al., 2002, 2003; Stoica et al., 2001; Trastour, Bartolini, & Gonzalez-Castillo, 2001), and access control (Bell & LaPadula, 1976; Johnston, Mudumbai, & Thompson, 1998; Pearlman et al., 2002; Sandhu & Samarati, 1994; Sandhu et al., 1996; Zhang & Parashar, 2003), the results either cannot be applied to COPCEs due to the reasons (1) through (4) discussed in the last section, or do not incorporate trustworthiness.

Service Specification Languages

To facilitate generation, publishing, discovery, and usage of services for data sharing, a service specification language is needed to specify the semantic description of the data to be shared, the access interfaces to be provided, and the access control policies for the shared data.

Currently, several service specification languages are widely used such as Web Service Description Language (WSDL) (W3C, 2001) and Business Process Execution Language for Web Services (BPEL4WS) (Andrews et al., 2003), or have become more popular such as OWL-S (formerly DAML-S) (W3C, 2004b). However, WSDL and BPEL4WS specify the interfaces of services and cannot specify the semantics of the shared data. Furthermore, for specifying security policies, WSDL and BPEL4WS rely on WS-Security (Atkinson et al., 2002) and WS-SecurityPolicy (Della-Libera et al., 2002) to satisfy the basic requirements of message integrity, confidentiality, and single-message authentication for Web services. But, it is very difficult to specify declarative access control policies using WS-Security and WS-SecurityPolicy. Other notable languages for specifying security policies include SAML (OASIS, 2004) and XACML (OASIS, 2003b). SAML is an XML-based security language for exchanging authentication and authorization information, but it puts too much burden on services to gather the evidence needed for policy decision. XACML provides schema and namespaces for access control policies, but its semantics are implicit and cause ambiguity when interpretations differ. Although OWL-S can be used to specify the semantic description of the data to be shared, it does not have the notations and structures to specify the access control policies for the shared data. Therefore, these service specification languages can hardly be used to specify services for trustworthy data sharing.

Service Discovery/Matchmaking

Before a published data service can be used by users, it must first be found by users. The process of finding a published data service based on users' requests is known as *service discovery* or *matchmaking*. To facilitate effective data sharing in COPCEs and ensure trustworthiness of information systems, the service discovery in COPCEs must satisfy at least the following requirements:

1. **Intelligent:** The data service discovery must be intelligent; that is, the service discovery should be based on the service semantics rather than the service interface.
2. **Secure:** Access control must be enforced when the service discovery is performed to prevent services from being discovered by untrustworthy or malicious users.
3. **Lightweight:** The shared data service discovery must be lightweight to maximize the battery lifetime of mobile devices.
4. **Robust:** The service discovery must be fault-tolerant to avoid single point of failure.

So far, no service discovery techniques can satisfy all these requirements simultaneously. For Requirement (i), there are service discovery techniques (Paolucci et al., 2002, 2003; Trastour et al., 2001) to provide the capability of matching services' semantics. For example, the technique presented in Paolucci et al. (2003) makes use of DAML-S for semantic matching and performs discovery based on Gnutella P2P protocol (Gnutella). However, these techniques do not address Requirements (2) and (4). For Requirement (2), although some existing service discovery techniques such as secure Service Discovery Service (SDS) (Czerwinski et al., 1999) allow control over who can discover and access the services, they make the assumption that service providers can provide a list of the principles that are allowed to access the services, and the access control is based on such a list. This kind of access control is static and not suitable for COPCEs, in which users may be added or removed from the systems dynamically. For Requirements (3) and (4), existing service discovery techniques for Web services, such as Universal Description Discovery & Integration (UDDI) (OASIS, 2003a), DAML-S Matchmaker (Paolucci et al., 2002), and a matchmaking service developed by HP (Trastour et al., 2001), are based on centralized service registries. However, the centralized service registry will be the performance bottleneck when a large amount of service queries need to be processed and cause single point of failure. Hence, these approaches are not suitable for dynamic environments like COPCEs. Other service discovery techniques (Balazinska et al., 2002; Czerwinski et al., 1999; Paolucci et al., 2003; Stoica et al., 2001) developed for P2P computing and ubiquitous computing do not use centralized service registries; hence, they are more dynamic and fault tolerant. However, these techniques have large overhead in service discovery and cannot guarantee the successful discovery of requested services. Yau and Karim (2003, 2004a, b) introduced energy-efficient protocols for situation-aware object discovery in ubiquitous computing environments. Although the approaches by Yau and Karim are not intended for service-based COPCAs, they can be extended to service-based COPCAs.

Access Control

To ensure the proper usage of published data services, access control mechanisms are needed. Access control mechanisms are based on three types of models: Mandatory Access Control (MAC) models (Bell & LaPadula, 1976), Discretionary Access Control models (Sandhu & Samarati, 1994), and Role-Based Access Control (RBAC) models (Sandhu et al., 1996). These models cannot be applied to COPCAs directly because of the following reasons:

- MAC models are based on an assumption that access decisions cannot be decided by the object owner (NIST, 1994). The MAC-based system enforces the access policy over the wishes or intentions of the object owner. But, in COPCAs, the owner of shared data may require making access decisions on the provided data. As security labels of object and subject are relatively static in MAC models, it is difficult to use MAC models to enforce flexible dynamic security policies.
- DAC models need to maintain an Access Control List (ACL) or capability. Due to the large amount of services and clients in a COPCA, it will be very difficult to maintain ACL or capability lists in DAC models. Revoking the granted access rights will also be very difficult.
- RBAC models assume that the User-Role and Role-Permission assignments are relatively static, which may not be the case in the COPCA environments.

To enforce flexible security policies and make access control more usable, Thuraisingham and Ford (1995) introduced a flexible security constraint handling process in distributed database management systems by assigning security levels to the data based on content, context, and time. In COPCEs, data to be shared is often unstructured or semi-structured. Thus, generating flexible access control policies for sharing data based on situations in COPCEs needs further improvement.

Research Issues and Future Research Directions

Based on the discussions on the current state-of-the-art, we will identify the research issues and future research directions in the three aspects (service specification languages, service discovery, and access control) in this section.

Data Service Specification for Trustworthy Data Sharing in Service-Based COPCAs

Data service specification is very important for the operations of service-based COPCAs since all service discovery mechanisms require the knowledge of certain types of service descriptions. In order to achieve trustworthy data sharing in service-based COPCAs, trustworthiness constraints for data services must be included in service specifications. Such service specifications should include at least the following three items:

(a) The description of the data to be shared such as the name of the data source, semantics, type, owner, and location of the data.
(b) The access interfaces to be provided such as query, update, and subscription.
(c) The trustworthiness constraints for data services such as access control policies (to be discussed later), availability, survivability, and auditability of data services. In COPCEs, these constraints usually vary due to context/situation changes. Hence, the service specifications must also include situation-awareness specifications.

Existing Web service specification languages cannot specify these three items together. Therefore, new specification languages for services for trustworthy data sharing need to be developed. The same languages should also be usable by users to specify their service requests.

A fundamental issue in developing such specification languages is how to model these services for trustworthy data sharing in COPCAs. The current service model of W3C (W3C, 2004a) is not suitable for trustworthy data sharing in COPCAs because it does not include mobility and situation-awareness and has no formal representation for service semantics. A notable ongoing work in SDK WSMO (Web Service Modeling Ontology) working group focuses on the development of an ontology for describing various aspects of semantic Web services, including non-functional properties, capability, interfaces, and ontology mediators to be used by Web services (Fensel & Bussler, 2002; WSMO, 2004). WSMO allows formal definition of real-world semantics based on logical frameworks such as F-Logic (Kifer, Lausen, & Wu, 1995) and has identified a set of non-functional properties, including security, reliability, and robustness for semantic Web services. Machine-understandable specification languages can be developed based on WSMO. These languages will enable the automation of service discovery and service composition. However, current WSMO lacks detailed definitions for the non-functional properties and does not consider mobility and situation-awareness either.

Therefore, to develop specification languages for services for trustworthy data sharing in COPCAs, we need to do the following: (1) incorporate mobility and situation-awareness into existing service models, (2) identify a set of trustworthiness constraints for data services and incorporate the models for these trustworthiness constraints into existing service models, and (3) develop machine-understandable languages based on the new service models.

Data Service Discovery for Trustworthy Data Sharing in Service-Based COPCAs

We have previously identified four requirements of service discovery for trustworthy data sharing in COPCAs: (1) intelligent, (2) secure, (3) lightweight, and (4) robust. As mentioned before, no existing service discovery techniques can satisfy all four requirements. The following research issues need to be addressed in order to develop service discovery mechanisms that can satisfy these requirements:

1. **Efficient distributed trust management for secure service discovery:** In COPCAs, users in the same group may be in geographically dispersed locations and need to communicate among themselves through public communication networks. Hence, the service query and response must both be secured to avoid being intercepted and manipulated by malicious agents. Furthermore, access control needs to be considered in service discovery to prevent the system from revealing confidential data services to users without proper privileges. In addition, services need to be authenticated to ensure the trustworthiness of shared data.

   As mentioned before, existing service discovery techniques require centralized certificate management and access control. However, in COPCAs, users may come from various organizations, each of which may have different security policies and require different certificates. Furthermore, centralized certificate management and access control are vulnerable to denial-of-service attacks. Therefore, new techniques for distributed trust management need to be developed and used in data service discovery. It should be noted that introducing security mechanisms into data service discovery will result in larger overhead in service discovery, which is a critical issue to be considered for mobile/embedded devices running on battery power.

2. **Reliability and robustness of service discovery:** As discussed before, service discovery techniques not using centralized registries or indexes are more suitable for COPCAs because they are more dynamic and more robust (avoid single point of failure). This type of service discovery techniques is normally based on multi-casting service queries or creating a direct mapping of services with hosts (e.g., DHT-based service discovery). The latter has better performance but is not suitable for COPCAs since it requires a uniform service-to-host mapping mechanism across the entire system and needs to have the capability to deploy services to different hosts according to the mapping, which is unrealistic in COPCAs involving multiple organizations. The service discovery techniques based on multi-casting service queries can be used in such an environment, but they are not reliable in the sense that they cannot guarantee the discovery of a requested service even if the service exists. However, for COPCAs, we believe that reliable service discovery techniques can be developed based on reliable group communication mechanisms. But, there will be a tradeoff between reliability and performance since reliable group communication will result in more overhead.
3. **Intelligence vs. efficiency:** As mentioned before, several approaches exist to provide the capability of matching services' semantics to enable intelligent service discovery. The size of service queries in these approaches is bigger than other non-semantic service discoveries since additional semantic information needs to be carried in service requests. However, in COPCEs, devices often communicate over wireless networks and rely on battery power. To what extent the bigger size of service queries will affect the resource consumption (network bandwidth, battery power) is not clear and needs to be investigated. If the size of service request has a big impact on the resource consumption, new data structures and compression algorithms need to be developed to effectively reduce the size of service requests.

Flexible and Adaptive Access Control to Data Services

There is an inherent tradeoff between security and performance. We should not make the system unusable or unmanageable in order to achieve security (Thuraisingham & Ford, 1995). One major research issue here is how to enforce flexible and adaptable access control policies for ensuring security with acceptable performance. The access control mechanism for COPCAs should address the dynamicity of the user activities and the environments of COPCAs. For example, Alice and Bob are two researchers working together on a nuclear research project. When Alice and Bob are in a secure room passed by security inspection, Alice should be allowed to access Bob's shared data services. When Alice is traveling outside, she should not be allowed to access Bob's shared data services.

Context-based access control systems have been studied extensively to provide adaptable access control for distributed and dynamic systems, especially pervasive computing systems. There are two ways to consider context in access control. One is to consider context values as constraints on access right granting. For example, Kumar, Karnik, and Chafle (2002) presented an extended role-based access control (RBAC) model, which includes context filters during the definition of a role to make RBAC sensitive to the context of an attempted operation. Extensive research on temporal constraints has been done on role-based access control (Bertino, Bonatti, & Ferrari, 2001; Joshi, 2003). The other way is to consider context values as principals in access control (Corradi, Montanari, & Tibaldi, 2004). Access rights are assigned to context values, and if the user/system has certain context value, then the user will have the access rights assigned to those context values. However, the dynamicity of user activity and the multiple domain policy interactions (Bell, 1994) have not been addressed. Therefore, to provide flexible and adaptive access control for COPCAs, we need to address the following research problems:

1. Develop a security policy ontology based OWL-S and integrate the dynamicity of user activity with the dynamicity of computing environments to incorporate situation-aware constraints in existing access control models. It will allow users to specify and enforce more flexible and adaptable access policies for COPCAs.

2. Develop a situation-aware access policy language based on the OWL-S based security policy ontology to specify access policies for data sharing. Such an OWL-S based language will enhance semantic interoperability of access policies across organizational boundaries.
3. Provide mechanisms for delegation, and policy confliction detection and resolution for sharing data in collaborative device groups.

Summary

In this chapter, a brief overview of data sharing and its trustworthiness requirements in COPCAs has been presented. Due to limited space, our discussions have focused on the two important aspects requiring most investigations: *data service specification and discovery for trustworthy data sharing,* and *flexible and adaptive access control to data services in service-based COPCAs*. The current state-of-the-art research issues and future research directions for *these two aspects* have been presented. The long-term goal of this research should be to develop techniques to enable dynamic, efficient, secure, and reliable data sharing among groups of users in pervasive computing environments to facilitate effective ad hoc group collaboration.

Acknowledgment

This work is supported in part by the National Science Foundation under the grant number IIS-0430565. The author would like to thank Dazhi Huang and Yisheng Yao of Arizona State University for many valuable discussions of the results presented in this chapter.

References

Andrews, T., et al. (2003, May). Business process execution language for Web services version 1.1. Retrieved May 7, 2005, from *http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/*

Atkinson, B., et al. (2002, April). Web services security v1.0. Retrieved May 7, 2005, from *http://www-106.ibm.com/developerworks/webservices/library/ws-secure/*

Balazinska, M., Balakrishnan, H., & Karger, D. (2002). INS/Twine: A scalable peer-to-peer architecture for intentional resource discovery. *Proceedings of the 1st International Conference on Pervasive Computing,* August (pp. 195-210).

Bell, D.E. (1994). Modeling the "Multipolicy Machine". *Proceedings of the New Security Paradigms Workshop*, August 3-5, Little Compton, RI, (pp. 2-9).

Bell, D.E., & LaPadula, L.J. (1976, May). *Secure computer system: Unified exposition and multics interpretation* (Tech. Rep. No. ESD-TR-75-306). Mitre Corporation.

Berket, K., & Agarwal, D. (2003). Enabling secure ad-hoc collaboration. *Proceedings of the Workshop on Advanced Collaborative Environments*, June. Retrieved May 7, 2005, from *http://www-itg.lbl.gov/Collaboratories/Publications/Karlo-WACE-2003-final.pdf*

Bertino, E., Bonatti, P.A., & Ferrari, E. (2001). TRBAC: A temporal role-based access control model. *ACM Transactions on Information Systems Security, 4*(3), 191-233.

Chen, G., & Kotz, D. (2003). Context-sensitive resource discovery. *Proceedings of the 1st International Conference on Pervasive Computing and Communications (PerCom'03),* March (pp. 243-252).

Chiang, T.C., & Huang, Y.M. (2003). Group keys and the multicast security in ad hoc networks. *Proceedings of the International Conference on Parallel Processing Workshop*, October (pp. 385-390).

Corradi, A., Montanari, R., & Tibaldi, D. (2004). Context-based access control for ubiquitous service provisioning. *Proceedings of the 28th Annual International Computer Software and Application Conference* (pp. 444-451).

Czerwinski, S., et al. (1999). An architecture for a secure service discovery service. *Proceedings of the 5th Annual International Conference on Mobile Computing and Networks (MobiCom'99)* (pp. 24-35).

Della-Libera, G., et al. (2002, December). Web service security policy. Retrieved May 7, 2005, from *http://www-106.ibm.com/developerworks/library/ws-secpol/*

Fensel, D., & Bussler, C. (2002). The Web service modeling framework WSMF. *Electronic Commerce Research and Applications, 1*(2), 113-137.

Gnutella. The Gnutella protocol specification v0.4. Retrieved May 7, 2005, from *http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf*

Guy, R., Reiher, P., Ratner, D., Gunter, M., Ma, W., & Popek, G. (1998). Rumor: Mobile data access through optimistic peer-to-peer replication. *Proceedings of the Workshop on Mobile Data Access,* 254-265.

Johnston, W., Mudumbai, S., & Thompson, M. (1998). Authorization and attribute certificates for widely distributed access control. *Proceedings of the IEEE 7th International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises* (pp. 340-345).

Joshi, J. (2003). *A generalized temporal role based access control model for developing secure systems.* Unpublished Ph D dissertation, Purdue University, West Lafayette, Indiana.

Kifer, M., Lausen, G., & Wu, J. (1995). Logical foundations of object oriented and frame-based languages. *Journal of the ACM, 42*(4), 741-843.

Kumar, A., Karnik, N., & Chafle, G. (2002). Context sensitivity in role-based access control. *ACM SIGOPS Operating Systems Review, 36*(3), 53-66.

Kyasanur, P., & Vaidya, N.H. (2002, August). *Detection and handling of MAC layer misbehavior in wireless networks* (Tech. Rep.). University of Illinois at Urbana-Champaign, Coordinated Science Laboratory. Retrieved May 7, 2005, from *http://citeseer.nj.nec.com/kyasanur02detection.html*

Lazos, L., & Poovendran, R. (2003). Energy-aware secure multicast communication in ad-hoc networks using geographic location information. *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing,* April (pp. 201-204).

NIST. (1994). Security in open systems. *NIST Special Publication SP800-7*. Retrieved May 7, 2005, from *http://csrc.nist.gov/publications/nistpubs/800-7/*

OASIS. (2003a). UDDI version 3.0.1. Retrieved May 7, 2005, from *http://uddi.org/pubs/uddi_v3.htm*

OASIS. (2003b). eXtensible Access Control Markup Language (XACML) version 1.0. Retrieved May 7, 2005, from *http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml*

OASIS. (2004, August). OASIS security assertion markup language (SAML) v2.0. Retrieved May 7, 2005, from *http://www.oasis-open.org/committees/download.php/8778/sstc-saml-conformance-2.0-cd-01.pdf*

Paolucci, M., Kawamura, T., Payne, T.R., & Sycara, K. (2002). Semantic matching of Web services capabilities. *Proceedings of the 1st International Semantic Web Conference (ISWC'02),* June (pp. 333-347).

Paolucci, M., Sycara, K., Nishimura, T., & Srinivasan, N. (2003). Using DAML-S for P2P discovery. *Proceedings of the International Conference on Web Services (ICWS'03)* June (pp. 203-207).

Pearlman, L., Welsh, V., Foster, I., Kesselman, C., & Tuecke, S. (2002). A community authorization service for group collaboration. *Proceedings of the IEEE 3rd International Workshop on Policies for Distributed Systems and Networks* (pp. 50-59).

Ratner, D., Popek, G.J., & Reiher, P. (1999). Roam: A scalable replication system for mobile computing. *Mobility in Databases and Distributed Systems,* 96-104.

Sandhu, R., Coyne, E.J., Feinstein, H.L., & Youman, C.E. (1996). Role based access control models. *IEEE Computer, 29*(2), 38-47.

Sandhu, R., & Samarati, P. (1994). Access control: Principles and practice. *IEEE Communications Magazine, 32*(9), 40-48.

Stoica, I., Morris, R., Karger, D., Kaashoek, F., & Balakrishnan, H. (2001, August). Chord: A scalable peer-to-peer lookup service for Internet applications. *Proceedings of the ACM SIGCOMM'01* (pp. 149-160).

Thuraisingham, B., & Ford, W. (1995, April). Security constraints processing in multilevel secure distributed database management system. *IEEE Transactions on Knowledge and Data Engineering, 7*(2), 274-293.

Trastour, D., Bartolini, C., & Gonzalez-Castillo, J. (2001). A Semantic Web approach to service description for matchmaking of services. *Proceedings of the 1st International Semantic Web Working Symposium,* July. Retrieved May 7, 2005, from *http://citeseer.nj.nec.com/trastour01semantic.html*

W3C. (2001). Web service description language (WSDL) 1.1. Retrieved May 7, 2005, from *http://www.w3.org/TR/wsdl*

W3C. (2004a). Web services architecture. Retrieved May 7, 2005, from *http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/*

W3C. (2004b). OWL Web Ontology Language: Overview. Retrieved May 7, 2005, from *http://www.w3.org/TR/owl-features*

WSMO. (2004, March). Web service modeling ontology (WSMO) standard. Retrieved May 7, 2005, from *http://www.wsmo.org/2004/d2/v0.2/20040306/*

Yau, S.S., & Karim, F. (2003, November). An energy-efficient object discovery protocol for context-sensitive middleware for ubiquitous computing. *IEEE Transactions on Parallel and Distributed Systems, 14*(11), 1074-1084.

Yau, S.S., & Karim, F. (2004a, February). A context-sensitive middleware-based approach to dynamically integrating mobile devices into computational infrastructures. *Journal of Parallel and Distributed Computing, 64*(2), 301-317.

Yau, S.S., & Karim, F. (2004b). An adaptive middleware for context-sensitive communications for real-time applications in ubiquitous computing environments. *Real-Time Systems, 26*(1), 29-61.

Yau, S.S., Wang, Y., & Karim, F. (2002). Development of situation-aware application software for ubiquitous computing environments. *Proceedings of the 26th IEEE International Computer Software and Applications Conference (COMPSAC 2002),* August (pp. 233-238).

Yau, S.S., & Zhang, X. (2004, December). A middleware service for secure group communication in mobile ad hoc networks. *The Journal of Systems and Software, 76*, 29-43.

Zhang, Y., Lee, W., & Huang, Y. (2003, September). Intrusion detection techniques for mobile wireless networks. *ACM/Kluwer Wireless Networks Journal (ACM WINET), 9*(5), 545-556.

Zhang, G., & Parashar, M. (2003). Dynamic context-aware access control for grid applications. *Proceedings of the 4th International Workshop on Grid Computing (Grid 2003),* November (pp. 101-108).

Chapter XIV

Privacy-Preserving Data Mining on the Web: Foundations and Techniques

Stanley R. M. Oliveira, Embrapa Informática Agropecuária, Brazil

Osmar R. Zaïane, University of Alberta, Edmonton, Canada

Abstract

Privacy-preserving data mining (PPDM) is one of the newest trends in privacy and security research. It is driven by one of the major policy issues of the information era—the right to privacy. This chapter describes the foundations for further research in PPDM on the Web. In particular, we describe the problems we face in defining what information is private in data mining. We then describe the basis of PPDM including the historical roots, a discussion on how privacy can be violated in data mining, and the definition of privacy preservation in data mining based on users' personal information and information concerning their collective activities. Subsequently, we introduce a taxonomy of the existing PPDM techniques and a discussion on how these techniques are applicable to Web-based *applications. Finally, we suggest some privacy requirements that are related to industrial initiatives and point to some technical challenges as future research trends in PPDM on the Web.*

Introduction

Analyzing what right to privacy means is fraught with problems, such as whether the exact definition of privacy constitutes a fundamental right and whether people are and should be concerned with it. Several definitions of privacy have been given, and they vary according to context, culture, and environment. For instance, in a seminal paper, Warren and Brandeis (1890) defined privacy as "the right to be alone". Later on, Westin (1967) defined privacy as "the desire of people to choose freely under what circumstances and to what extent they will expose themselves, their attitude, and their behavior to others". Schoeman (1984) defined privacy as "the right to determine what (personal) information is communicated to others" or "the control an individual has over information about himself or herself". More recently, Garfinkel (2001) stated that "privacy is about self-possession, autonomy, and integrity". On the other hand, Rosenberg (2000) argues that privacy may not be a right after all but a taste: "If privacy is in the end a matter of individual taste, then seeking a moral foundation for it—beyond its role in making social institutions possible that we happen to prize—will be no more fruitful than seeking a moral foundation for the taste for truffles".

The above definitions suggest that, in general, privacy is viewed as a social and cultural concept. However, with the ubiquity of computers and the emergence of the Web, privacy has also become a digital problem. With the Web revolution and the emergence of data mining, privacy concerns have posed technical challenges fundamentally different from those that occurred before the information era. In the information technology era, privacy refers to the right of users to conceal their personal information and have some degree of control over the use of any personal information disclosed to others (Cockcroft & Clutterbuck, 2001).

In the context of data mining, the definition of privacy preservation is still unclear, and there is very little literature related to this topic. A notable exception is the work presented in Clifton, Kantarcioglu, and Vaidya (2002), in which PPDM is defined as "getting valid data mining results without learning the underlying data values". However, at this point, each existing PPDM technique has its own privacy definition. Our primary concern about PPDM is that mining algorithms are analyzed for the side effects they incur in data privacy. We define PPDM as the dual goal of meeting privacy requirements and providing valid data mining results.

The Basis of Privacy-Preserving Data Mining

Historical Roots

The debate on PPDM has received special attention as data mining has been widely adopted by public and private organizations. We have witnessed three major landmarks that characterize the progress and success of this new research area: *the conceptive landmark*, *the deployment landmark*, and *the prospective landmark*. We describe these landmarks as follows:

The *conceptive landmark* characterizes the period in which central figures in the community, such as O'Leary (1991, 1995), Piatetsky-Shapiro (1995), and others (Klösgen, 1995; Clifton & Marks, 1996), investigated the success of knowledge discovery and some of the important areas where it can conflict with privacy concerns. The key finding was that knowledge discovery can open new threats to informational privacy and information security if not done or used properly. The *deployment landmark* is the current period in which an increasing number of PPDM techniques have been developed and published in refereed conferences. The information available today is spread over countless papers and conference proceedings. The results achieved in the last years are promising and suggest that PPDM will achieve the goals that have been set for it.

The *prospective landmark* is a new period in which directed efforts toward standardization occur. At this stage, there is no consent about what privacy preservation means in data mining. In addition, there is no consensus on privacy principles, policies, and requirements as a foundation for the development and deployment of new PPDM techniques. The excessive number of techniques is leading to confusion among developers, practitioners, and others interested in this technology. One of the most important challenges in PPDM now is to establish the groundwork for further research and development in this area.

Privacy Violation in Data Mining

Understanding privacy in data mining requires understanding how privacy can be violated and the possible means for preventing privacy violation. In general, one major factor contributes to privacy violation in data mining: the misuse of data.

Users' privacy can be violated in different ways and with different intentions. Although data mining can be extremely valuable in many applications (e.g., business, medical analysis, etc.), it can also, in the absence of adequate safeguards, violate informational privacy. Privacy can be violated if personal data are used for other purposes subsequent to the original transaction between an individual and an organization when the information was collected (Culnan, 1993).

One of the sources of privacy violation is called data magnets (Rezgui, Bouguettaya, & Eltoweissy, 2003). Data magnets are techniques and tools used to collect personal data. Examples of data magnets include explicitly collecting information through online registration, identifying users through IP addresses, software downloads that require registration, and indirectly collecting information for secondary usage. In many cases, users may or may not be aware that information is collected or do not know how that information is collected. In particular, collected personal data can be used for secondary usage largely beyond the users' control and privacy laws. This scenario has led to an uncontrollable privacy violation, not because of data mining itself, but fundamentally because of the misuse of data.

Defining Privacy for Data Mining

In general, privacy preservation occurs in two major dimensions: users' personal information and information concerning their collective activities. We refer to the former as individual privacy preservation and the latter as collective privacy preservation, which is related to corporate privacy in Clifton et al. (2002).

- **Individual privacy preservation:** The primary goal of data privacy is the protection of personally identifiable information. In general, information is considered personally identifiable if it can be linked, directly or indirectly, to an individual person. Thus, when personal data are subjected to mining, the attribute values associated with individuals are private and must be protected from disclosure. Miners are then able to learn from global models rather than from the characteristics of a particular individual.

- **Collective privacy preservation:** Protecting personal data may not be enough. Sometimes, we may need to protect against learning sensitive knowledge representing the activities of a group. We refer to the protection of sensitive knowledge as collective privacy preservation. The goal here is quite similar to the one for statistical databases, in which security control mechanisms provide aggregate information about groups (population) and, at the same time, prevent disclosure of confidential information about individuals. However, unlike in the case for statistical databases, another objective of collective privacy preservation is to protect sensitive knowledge that can provide competitive advantage in the business world.

In the case of collective privacy preservation, organizations have to cope with some interesting conflicts. For instance, when personal information undergoes analysis processes that produce new facts about users' shopping patterns, hobbies, or preferences, these facts could be used in recommender systems to predict or affect their future shopping patterns. In general, this scenario is beneficial to both users and organizations. However, when organizations share data in a collaborative project, the goal is not only to protect personally identifiable information but also sensitive knowledge represented by some strategic patterns.

Characterizing Scenarios of Privacy Preservation on the Web

In this section, we describe two real-life motivating examples in which PPDM poses different constraints:

- **Scenario 1:** Suppose we have a server and many clients in which each client has a set of sold items (e.g., books, movies, etc.). The clients want the server to gather statistical information about associations among items in order to provide recommendations to the clients. However, the clients do not want the server to know some strategic patterns (also called sensitive association rules). In this context, the clients represent companies, and the server is a recommendation system for an e-commerce application, for example, fruit of the clients collaboration. In the absence of rating, which is used in collaborative filtering for automatic recommendation building, association rules can be effectively used to build models for online recommendation. When clients send their frequent itemsets or association rules to the server, it must protect the sensitive itemsets according to some specific policies. The server then gathers statistical information from the non-sensitive itemsets and recovers from them the actual associations. How can these companies benefit from such collaboration by sharing association rules while preserving some sensitive association rules?

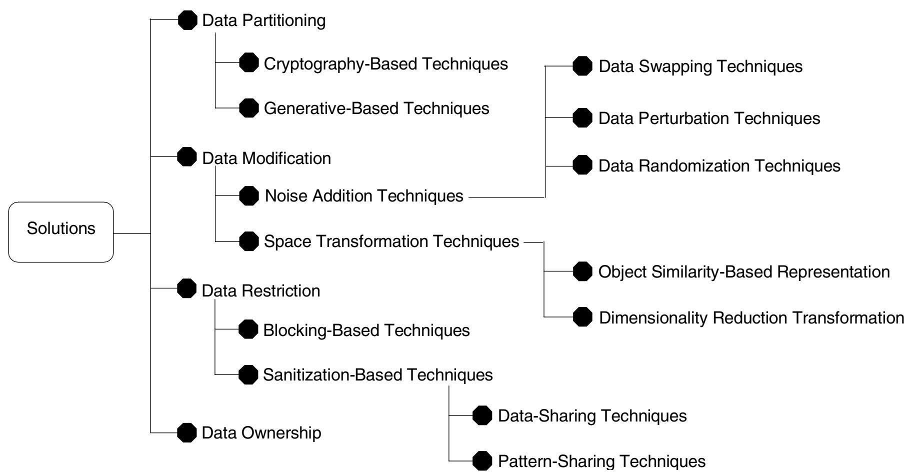
- **Scenario 2:** Two organizations, an Internet marketing company, and an online retail company have datasets with different attributes for a common set of individuals. These organizations decide to share their data for clustering to the optimal customer targets so as to maximize return on investments. How can these organizations learn about their clusters using each other's data without learning anything about the attribute values of each other?

Note that the above scenarios describe different privacy preservation problems. Each scenario poses a set of challenges. For instance, Scenario 1 is a typical example of collective privacy preservation, while Scenario 2 refers to individuals' privacy preservation.

A Taxonomy of Existing PPDM Techniques

In this section, we classify the existing PPDM techniques in the literature into four major categories: data partitioning, data modification, data restriction, and data ownership, as can be seen in Figure 1.

Figure 1. A taxonomy of PPDM techniques



Data Partitioning Techniques

Data partitioning techniques have been applied to some scenarios in which the databases available for mining are distributed across a number of sites, with each site only willing to share data mining results, not the source data. In these cases, the data are distributed either horizontally or vertically. In a horizontal partition, different entities are described with the same schema in all partitions, while in a vertical partition the attributes of the same entities are split across the partitions. The existing solutions can be classified into Cryptography-Based Techniques and Generative-Based Techniques.

- **Cryptography-Based Techniques:** In the context of PPDM over distributed data, cryptography-based techniques have been developed to solve problems of the following nature: two or more parties want to conduct a computation based on their private inputs. The issue here is how to conduct such a computation so that no party knows anything except its own input and the results. This problem is referred to as the Secure Multi-Party Computation (SMC) problem (Goldreich, Micali, & Wigderson, 1987). The technique proposed in Lindell and Pinkas (2000) addresses privacy-preserving classification, while the techniques proposed in

Kantarciolu and Clifton (2002) and Vaidya and Clifton (2002) address privacy-preserving association rule mining, and the technique in Vaidya and Clifton (2003) addresses privacy-preserving clustering.

- **Generative-Based Techniques:** These techniques are designed to perform distributed mining tasks. In this approach, each party shares just a small portion of its local model which is used to construct the global model. The existing solutions are built over horizontally partitioned data. The solution presented in Veloso, Meira, Parthasarathy, and Carvalho (2003) addresses privacy-preserving frequent itemsets in distributed databases, whereas the solution in Meregu and Ghosh (2003) addresses privacy-preserving distributed clustering using generative models.

Data Modification Techniques

Data modification techniques modify the original values of a database that need to be shared, and in doing so, privacy preservation is ensured. The transformed database is made available for mining and must meet privacy requirements without losing the benefit of mining. In general, data modification techniques aim at finding an appropriate balance between privacy preservation and knowledge disclosure. Methods for data modification include noise addition techniques and space transformation techniques.

- **Noise Addition Techniques:** The idea behind noise addition techniques for PPDM is that some noise (e.g., information not present in a particular tuple or transaction) is added to the original data to prevent the identification of confidential information relating to a particular individual. In other cases, noise is added to confidential attributes by randomly shuffling the attribute values to prevent the discovery of some patterns that are not supposed to be discovered. We categorize noise addition techniques into three groups: (1) data swapping techniques that interchange the values of individual records in a database (Estivill-Castro & Brankovic, 1999); (2) data distortion techniques that perturb the data to preserve privacy, and the distorted data maintain the general distribution of the original data (Agrawal & Srikant, 2000); and (3) data randomization techniques which allow one to perform the discovery of general patterns in a database with error bound, while protecting individual values. Like data swapping and data distortion techniques, randomization techniques are designed to find a good compromise between privacy protection and knowledge discovery (Agrawal & Gehrke, 2002; Evfimievski, Srikant, Rizvi, & Haritsa, 2002; Zang, Wang, & Zhao, 2004).

- **Space Transformation Techniques:** These techniques are specifically designed to address privacy-preserving clustering. These techniques are designed to protect the underlying data values subjected to clustering without jeopardizing the similarity between objects under analysis. Thus, a space transformation technique must not only meet privacy requirements but also guarantee valid clustering results. We categorize space transformation techniques into two major groups: (1) object similarity-based representation relies on the idea behind the similarity between objects; that is, a data owner could share some data for clustering analysis by simply computing the dissimilarity matrix (matrix of distances) between the objects and then sharing such a matrix with a third party. Many clustering algorithms in the literature operate on a dissimilarity matrix (Han & Kamber, 2001). This solution is simple to implement and is secure but requires a high communication cost (Oliveira & Zaïane, 2004); (2) dimensionality reduction-based transformation can be used to address privacy-preserving clustering when the attributes of objects are available either in a central repository or vertically partitioned across many sites. By reducing the dimensionality of a dataset to a sufficiently small value, one can find a tradeoff between privacy, communication cost, and accuracy. Once the dimensionality of a database is reduced, the released database preserves (or slightly modifies) the distances between data points. In tandem with the benefit of preserving the similarity between data points, this solution protects individuals' privacy since the attribute values of the objects in the transformed data are completely different from those in the original data (Oliveira & Zaïane, 2004).

Data Restriction Techniques

Data restriction techniques focus on limiting access to mining results through either generalization or suppression of information (e.g., items in transactions, attributes in relations), or even by blocking access to some patterns that are not supposed to be discovered. Such techniques can be divided into two groups: Blocking-based techniques and Sanitization-based techniques.

- **Blocking-Based Techniques:** These techniques aim at hiding some sensitive information when data are shared for mining. The private information includes sensitive association rules and classification rules that must remain private. Before releasing the data for mining, data owners must consider how much information can be inferred or calculated from large databases and must look for ways to minimize the leakage of such information. In general, blocking-based techniques are feasible to recover patterns less frequent than originally since sensitive information is either suppressed or replaced with unknowns to preserve privacy. The techniques in Johnsten and Raghavan (2001) address privacy preservation in classification, while the techniques in Johnsten and Raghavan (2002) and Saygin, Verykios, and Clifton (2001) address privacy-preserving association rule mining.
- **Sanitization-Based Techniques:** Unlike blocking-based techniques that hide sensitive information by replacing some items or attribute values with unknowns, sanitization-based techniques hide sensitive information by strategically suppressing some items in transactional databases, or even by generalizing information to preserve privacy in classification. These techniques can be categorized into two major groups: (1) data-sharing techniques in which the sanitization process acts on the data to remove or hide the group of sensitive association rules that contain sensitive knowledge. To do so, a small number of transactions that contain the sensitive rules have to be modified by deleting one or more items from them or even adding some noise, that is, new items not originally present in such transactions (Dasseni, Verykios, Elmagarmid, & Bertino, 2001; Oliveira & Zaïane, 2002, 2003a, 2003b; Verykios et al., 2004); and (2) pattern-sharing techniques in which the sanitizing algorithm acts on the rules mined from a database, instead of the data itself. The existing solution removes all sensitive rules before the sharing process and blocks some inference channels (Oliveira, Zaïane, & Saygin, 2004). In the context of predictive modeling, a framework was proposed in Iyengar (2002) for preserving the anonymity of individuals or entities when data are shared or made public.

Data Ownership Techniques

Data ownership techniques can be applied to two different scenarios: (1) to protect the ownership of data by people about whom the data were collected

(Felty & Matwin, 2002). The idea behind this approach is that a data owner may prevent the data from being used for some purposes and allow them to be used for other purposes. To accomplish that, this solution is based on encoding permissions on the use of data as theorems about programs that process and mine the data. Theorem-proving techniques are then used to guarantee that these programs comply with the permissions; and (2) to identify the entity that receives confidential data when such data are shared or exchanged (Mucsi-Nagy & Matwin, 2004). When sharing or exchanging confidential data, this approach ensures that no one can read confidential data except the receiver(s). It can be used in different scenarios, such as statistical or research purposes, data mining, and online business-to-business (B2B) interactions.

Are These Techniques Applicable to Web Data?

After describing the existing PPDM techniques, we now move on to analyze which of these techniques are applicable to Web data. To do so, hereinafter we use the following notation:

- **WDT:** These techniques are designed essentially to support Web usage mining; that is, the techniques address Web data applications only. We refer to these techniques as Web Data Techniques (WDT).
- **GPT:** These techniques can be used to support both public data release and Web-based applications. We refer to these techniques as General Purpose Techniques (GPT).

(a) **Cryptography-Based Techniques:** These techniques can be used to support business collaboration on the Web. Scenario 2 (in section: The Basis of Privacy-Preserving Data Mining) is a typical example of a Web-based application which can be addressed by cryptography-based techniques. Other applications related to e-commerce can be found (Srivastava, Cooley, Deshpande, & Tan, 2000; Kou & Yesha, 2000). Therefore, such techniques are classified as WDT.

(b) **Generative-Based Techniques:** These techniques can be applied to scenarios in which the goal is to extract useful knowledge from large, distributed data repositories. In these scenarios, the data cannot be directly centralized or unified as a single file or database either due to legal, proprietary, or technical restrictions. In general, generative-based techniques are designed to support distributed Web-based applications.

(c) **Noise Addition Techniques:** These techniques can be categorized as GPT. For instance, data swapping and data distortion techniques are used for public data release, while data randomization could be used to build models for online recommendations (Zang et al., 2004). Scenario 1 (in section: The Basis of Privacy-Preserving Data Mining) is a typical example of an online recommendation system.

(d) **Space Transformation Techniques:** These are general purpose techniques (GPT). These techniques could be used to promote social benefits as well as to address applications on the Web (Oliveira & Zaïane, 2004). An example of social benefit occurs, for instance, when a hospital shares some data for research purposes (e.g., cluster of patients with the same disease). Space transformation techniques can also be used when the data mining process is outsourced or even when the data are distributed across many sites.

(e) **Blocking-Based Techniques:** In general, these techniques are applied to protect sensitive information in databases. They could be used to simulate an access control in a database in which some information is hidden from users who do not have the right to access it. However, these techniques can also be used to suppress confidential information before the release of data for mining. We classify such techniques as GPT.

(f) **Sanitization-Based Techniques:** Like blocking-based techniques, sanitization-based techniques can be used by statistical offices who publish sanitized versions of data (e.g., census problem). In addition, sanitization-based techniques can be used to build models for online recommendations as described in Scenario 1 (in section: The Basis of Privacy-Preserving Data Mining).

(g) **Data Ownership Techniques:** These techniques implement a mechanism enforcing data ownership by the individuals to whom the data belongs. When sharing confidential data, these techniques can also be used to ensure that no one can read confidential data except the receiver(s) that are authorized to do so. The most evident applications of such techniques are related to Web mining and online business-to-business (B2B) interactions.

Table 1. A summary of the PPDM techniques and their relationship with Web data

PPDM Techniques Category	**Category**
Cryptography-Based Techniques	WDT
Generative-Based Techniques	WDT
Noise Addition Techniques	GPT
Space Transformation Technique	GPT
Blocking-Based Techniques	GPT
Sanitization-Based Techniques	GPT
Data Ownership Techniques	WDT

Table 1 shows a summary of the PPDM techniques and their relationship with Web data applications.

Requirements for Technical Solutions

Requirements for the Development of Technical Solutions

Ideally, a technical solution for a PPDM scenario would enable us to enforce privacy safeguards and to control the sharing and use of personal data. However, such a solution raises some crucial questions:

- What levels of effectiveness are in fact technologically possible, and what corresponding regulatory measures are needed to achieve these levels?
- What degrees of privacy and anonymity must be sacrificed to achieve valid data mining results?

These questions cannot have yes-no answers but involve a range of technological possibilities and social choices. The worst response to such questions is to ignore them completely and not pursue the means by which we can eventually provide informed answers. The above questions can be, to some extent, addressed if we provide some key requirements to guide the development of technical solutions.

The following keywords are used to specify the extent to which an item is a requirement for the development of technical solutions to address PPDM:

- **Must**: this word means that the item is an absolute requirement;
- **Should**: this word means that valid reasons not to treat this item as a requirement may exist, but the full implications should be understood and the case carefully weighed before discarding this item.

(a) **Independence:** A promising solution for the problem of PPDM, for any specific data mining task (e.g., association rules, clustering, and classification), should be independent of the mining task algorithm.

(b) **Accuracy:** When it is possible, an effective solution should do better than trade off between privacy and accuracy on the disclosure of data mining results. Sometimes a tradeoff must be found as in Scenario 2 (in section: The Basis of Privacy-Preserving Data Mining).

(c) **Privacy Level:** This is also a fundamental requirement in PPDM. A technical solution must ensure that the mining process does not violate privacy up to a certain degree of security.

(d) **Attribute Heterogeneity:** A technical solution for PPDM should handle heterogeneous attributes (e.g., categorical and numerical).

(e) **Communication Cost:** When addressing data distributed across many sites, a technical solution should carefully consider issues of communication cost.

Requirements to Guide the Deployment of Technical Solutions

Information technology vendors in the near future will offer a variety of products which claim to help protect privacy in data mining. How can we evaluate and decide whether what is offered is useful? The non-existence of proper instruments to evaluate the usefulness and feasibility of a solution to address a PPDM scenario challenge is to identify the following requirements:

(a) **Privacy Identification:** We should identify what information is private. Is the technical solution aimed at protecting individual privacy or collective privacy?

(b) **Privacy Standards:** Does the technical solution comply with international instruments that state and enforce rules (e.g., principles and/or policies) for use of automated processing of private information?

(c) **Privacy Safeguards:** Is it possible to record what has been done with private information and be transparent with individuals about whom the private information pertains?

(d) **Disclosure Limitation:** Are there metrics to measure how much private information is disclosed? Since privacy has many meanings depending on the context, we may require a set of metrics to do so. What is most important is that we need to measure not only how much private information is disclosed but also the impact of a technical solution on the data and on valid mining results.

(e) **Update Match:** When a new technical solution is launched, two aspects should be considered: (1) the solution should comply with existing privacy principles and policies, and (2) in case of modifications to privacy principles and/or policies that guide the development of technical solutions, any release should consider these new modifications.

Future Research Trends

Preserving privacy on the Web has an important impact on many Web activities and Web applications. In particular, privacy issues have attracted a lot of attention due to the growth of e-commerce and e-business. These issues are further complicated by the global and self-regulatory nature of the Web.

Privacy issues on the Web are based on the fact that most users want to maintain strict anonymity on Web applications and activities. The easy access to information on the Web, coupled with the readily available personal data also make it easier and more tempting for interested parties (e.g., businesses and governments) to willingly or inadvertently intrude on individuals' privacy in unprecedented ways.

Clearly, privacy issues on Web data is an umbrella that encompasses many Web applications such as e-commerce, stream data mining, multimedia mining, among others. In this work, we focus on issues toward foundation for further research in PPDM on the Web because these issues will certainly play a significant role in the future of this area. In particular, a common framework for PPDM should be conceived, notably in terms of definitions, principles, policies, and requirements. The advantages of a framework of that nature are as follows: (a) a common framework will avoid confusing developers, practitioners, and many others interested in PPDM on the Web; (b) adoption of a common framework will inhibit inconsistent efforts in different ways and will enable vendors and developers to make solid advances in the future of research in PPDM on the Web.

The success of a framework of this nature can only be guaranteed if it is backed by a legal framework such as the Platform for Privacy Preferences (P3P) Project (Joseph & Faith, 1999). This project is emerging as an industry standard providing a simple, automated way for users to gain more control over the use of personal information on Web sites they visit.

The European Union has taken a lead in setting up a regulatory framework for Internet Privacy and has issued a directive that sets guidelines for processing and transfer of personal data (European Comission, 1998).

Conclusion

In this chapter, we have laid down the foundations for further research in the area of Privacy-Preserving Data Mining (PPDM) on the Web. Although our work described in this chapter is preliminary and conceptual in nature, it is a vital prerequisite for the development and deployment of new techniques. In particular, we described the problems we face in defining what information is private in data mining. We then described the basis of PPDM including the historical roots, a discussion on how privacy can be violated in data mining, and the definition of privacy preservation in data mining based on users' personal information and information concerning their collective activities. We also introduced a taxonomy of the existing PPDM techniques and a discussion on how these techniques are applicable to Web data. Subsequently, we suggested some desirable privacy requirements that are related to industrial initiatives. These requirements are essential for the development and deployment of technical solutions. Finally, we pointed to standardization issues as a technical challenge for future research trends in PPDM on the Web.

References

Agrawal, R., & Srikant, R. (2000). Privacy-preserving data mining. *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data* (pp. 439-450), Dallas, TX.

Clifton, C., Kantarciolu, M., & Vaidya, J. (2002). Defining privacy for data mining. *Proceedings of the National Science Foundation Workshop on Next Generation Data Mining* (pp. 126-133), Baltimore.

Clifton, C., & Marks, D. (1996). Security and privacy implications of data mining. *Proceedings of the Workshop on Data Mining and Knowledge Discovery* (pp. 15-19), Montreal, Canada.

Cockcroft, S., & Clutterbuck, P. (2001). Attitudes towards information privacy. *Proceedings of the 12th Australasian Conference on Information Systems,* Coffs Harbour, NSW, Australia.

Culnan, M.J. (1993). How did they get my name? An exploratory investigation of consumer attitudes toward secondary information. *MIS Quartely, 17*(3), 341-363.

Dasseni, E., Verykios, V.S., Elmagarmid, A.K., & Bertino, E. (2001). Hiding association rules by using confidence and support. *Proceedings of the 4th Information Hiding Workshop* (pp. 369-383), Pittsburgh, Pennsylvania.

Estivill-Castro, V., & Brankovic, L. (1999). Data swapping: Balancing privacy against precision in mining for logic rules. *Proceedings of the Data Warehousing and Knowledge Discovery DaWaK-99*, Florence, Italy (pp. 389-398).

European Comission. (1998). The directive on the protection of individuals with regard of the processing of personal data and on the free movement of such data. Retrieved May 7, 2005, from *http://www2.echo.lu*

Evfimievski, A., Srikant, R., Agrawal, R., & Gehrke, J. (2002). Privacy preserving mining of association rules. *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 217-228), Edmonton, AB, Canada.

Felty, A.P., & Matwin, S. (2002). Privacy-oriented data mining by proof checking. *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD)* (pp. 138-149), Helsinki, Finland.

Garfinkel, S. (2001). *Database nation: The death of the privacy in the 21st century*. Sebastopol, CA: O'Reilly & Associates.

Goldreich, O., Micali, S., & Wigderson, A. (1987). How to play any mental game: A completeness theorem for protocols with honest majority. *Proceedings of the 19th Annual ACM Symposium on Theory of Computing* (pp. 218-229). New York.

Han, J., & Kamber, M. (2001). *Data mining: Concepts and techniques.* San Francisco: Morgan Kaufmann.

Iyengar, V.S. (2002). Transforming data to satisfy privacy constraints. *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, AB, Canada (pp. 279-288).

Johnsten, T., & Raghavan, V.V. (2001). Security procedures for classification mining algorithms. *Proceedings of the 15th Annual IFIP WG 11.3 Working Conference on Database and Applications Security*, Niagara on the Lake, Ontario, Canada (pp. 293-309).

Johnsten, T., & Raghavan, V.V. (2002). A methodology for hiding knowledge in databases. *Proceedings of the IEEE ICDM Workshop on Privacy, Security, and Data Mining*, Maebashi City, Japan (pp. 9-17).

Joseph, R., & Faith, C.L. (1999). The platform for privacy preferences. *Communications of the ACM, 42*(2), 48-55.

Kantarciolu, M., & Clifton, C. (2002). Privacy-preserving distributed mining of association rules on horizontally partitioned data. *Proceedings of the ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, Madison, Wisconsin.

Klösgen, W. (1995). KDD: Public and private concerns. *IEEE EXPERT, 10*(2), 55-57.

Kou, W., & Yesha, Y. (2000). *Electronic commerce technology trends: Challenges and opportunities.* IBM Press Alliance Publisher: David Uptmor, IIR Publications, Inc.

Lindell, Y., & Pinkas, B. (2000). Privacy preserving data mining. *Crypto 2000, LNCS 1880*, 36-54.

Meregu, S., & Ghosh, J. (2003). Privacy-preserving distributed clustering using generative models. *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM'03)*, Melbourne, FL (pp. 211-218).

Mucsi-Nagy, A., & Matwin, S. (2004). Digital fingerprinting for sharing of confidential data. *Proceedings of the Workshop on Privacy and Security Issues in Data Mining*, Pisa, Italy (pp. 11-26).

O'Leary, D.E. (1991). Knowledge discovery as a threat to database security. In G. Piatetsky-Shapiro, & W.J. Frawley (Eds.), *Knowledge discovery in databases* (pp. 507-516). Menlo Park, CA: AAAI/MIT Press.

O'Leary, D.E. (1995). Some privacy issues in knowledge discovery: The OECD personal privacy guidelines. *IEEE EXPERT, 10*(2), 48-52.

Oliveira, S.R.M., & Zaïane, O.R. (2002). Privacy preserving frequent itemset mining. *Proceedings of the IEEE ICDM Workshop on Privacy, Security, and Data Mining*, Maebashi City, Japan (pp. 43-54).

Oliveira, S.R.M., & Zaïane, O.R. (2003a). Algorithms for balancing privacy and knowledge discovery in association rule mining. *Proceedings of the 7th International Database Engineering and Applications Symposium (IDEAS'03)*, Hong Kong, China (pp. 54-63).

Oliveira, S.R.M., & Zaïane, O.R. (2003b). Protecting sensitive knowledge by data sanitization. *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM'03)*, Melbourne, FL (pp. 613-616).

Oliveira, S.R.M., & Zaïane, O.R. (2004). Privacy-preserving clustering by object similarity-based representation and dimensionality reduction transformation. *Proceedings of the Workshop on Privacy and Security Aspects of Data Mining (PSADM'04) in conjunction with the 4th IEEE International Conference on Data Mining (ICDM'04)*, Brighton, UK (pp. 21-30).

Oliveira, S.R.M., Zaïane, O.R., & Saygin, Y. (2004). Secure association rule sharing. *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'04)*, Sydney, Australia (pp. 74-85).

Piatetsky-Shapiro, G. (1995). Knowledge discovery in personal data vs. privacy: A mini-symposium. *IEEE Expert, 10*(2), 46-47.

Rezgui, A., Bouguettaya, A., & Eltoweissy, M.Y. (2003). Privacy on the Web: Facts, challenges, and solutions. *IEEE Security & Privacy, 1*(6), 40-49.

Rizvi, S.J., & Haritsa, J.R. (2002). Maintaining data privacy in association rule mining. *Proceedings of the 28th International Conference on Very Large Data Bases,* Hong Kong, China.

Rosenberg, A. (2000). Privacy as a matter of taste and right. In E.F. Paul, F.D. Miller, & J. Paul (Eds.), *The right to privacy* (pp. 68-90). Cambridge University Press.

Saygin, Y., Verykios, V.S., & Clifton, C. (2001). Using unknowns to prevent discovery of association rules. *SIGMOD Record, 30*(4), 45-54.

Schoeman, F.D. (1984). *Philosophical dimensions of privacy.* Cambridge University Press.

Srivastava, J., Cooley, R., Deshpande, M., & Tan, P.-N. (2000). Web usage mining: Discovery and applications of usage patterns from Web data. *SIGKDD Explorations, 1*(2), 12-23.

Vaidya, J., & Clifton, C. (2002). Privacy preserving association rule mining in vertically partitioned data. *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, AB, Canada (pp. 639-644).

Vaidya, J., & Clifton, C. (2003). Privacy-preserving K-means clustering over vertically partitioned data. *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC (pp. 206-215).

Veloso, A.A., Meira, Jr., W., Parthasarathy, S., & Carvalho, M.B. (2003). Efficient, accurate and privacy-preserving data mining for frequent itemsets in distributed databases. *Proceedings of the 18th Brazilian Symposium on Databases*, Manaus, Brazil (pp. 281-292).

Verykios, V.S., Elmagarmid, A.K., Bertino, E., Saygin, Y., & Dasseni, E. (2004). Association rule hiding. *IEEE Transactions on Knowledge and Data Engineering, 16*(4), 434-447.

Warren, S.D., & Brandeis, L.D. (1890). The right to privacy. *Harvard Law Review, 4*(5), 193-220.

Westin, A.F. (1967). *The right to privacy.* New York: Atheneum.

Zang, N., Wang, S., & Zhao, W. (2004). A new scheme on privacy preserving association rule mining. *Proceedings of the 15th European Conference on Machine Learning and the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases,* Pisa, Italy.

About the Authors

Elena Ferrari (elena.ferrari@uninsubria.it) has been a professor of computer science at the University of Insubria at Como, Italy, since March 2001. From 1998 until January 2001, she was an assistant professor at the Department of Computer Science of the University of Milano (Italy). She received an MS in computer science from the University of Milano (1992) and a PhD in computer science from the same university (1998). She has also been a visiting researcher at the Department of Computer Science of George Mason University (USA), and at Rutgers University (USA). Her main research interests are Web security, access control and privacy, multimedia databases, and temporal databases. On these topics, she has published more than a hundred scientific publications in international journals and conference proceedings. Dr. Ferrari is the scientific director of the International Summer School on Information and System Security (ISS), annually held in Villa Olmo, Como (Italy). Dr. Ferrari has served as program chair of the 4th ACM Symposium on Access Control Models and Technologies (SACMAT'04), software demonstration chair of the 9th International Conference on Extending Database Technology (EDBT'04), co-chair of the 1st COMPSAC'02 Workshop on Web Security and Semantic Web, the 1st ECOOP Workshop on XML and Object Technology (XOT 2000), and the 1st ECOOP Workshop on Object-Oriented Databases. She has also served as program committee member of several international conferences and workshops. Dr. Ferrari is on the editorial board of the *VLDB Journal* and the *International Journal of Information Technology* (IJIT). She is a member of ACM and senior member of IEEE. For more information, visit *http://scienze-como.uninsubria.it/ferrari.*

Bhavani Thuraisingham (bhavani.thuraisingham@utdallas.edu) has recently joined the University of Texas at Dallas (USA) as a professor of computer science and director of the Cyber Security Research Center in the Erik Jonsson School of Engineering. She is a fellow of the IEEE (Institute for Electrical and Electronics Engineers) and the AAAS (American Association for the Advancement of Science). She received IEEE Computer Society's prestigious 1997 Technical Achievement Award for "outstanding and innovative contributions to secure data management." Thuraisingham's research in information security and information management has resulted in more than 60 journal articles, more than 200 refereed conference papers, and three U.S. patents. She is the author of seven books on data management, data mining, and data security including one on data mining for counter-terrorism. She has given more than 25 keynote presentations at various research conferences and has also given invited talks at the White House Office of Science and Technology Policy and at the United Nations on Data Mining for counter-terrorism. She serves (or has served) on editorial boards of several top research journals. Prior to joining UTD, Thuraisingham was an IPA (Intergovernmental Personnel Act) at the National Science Foundation from the MITRE Corporation. At NSF, she established the Data and Applications Security Program and co-founded the Cyber Trust theme and was involved in inter-agency activities in data mining for counter-terrorism. She has been at MITRE since January 1989 and has worked in MITRE's Information Security Center and was later a department head in Data and Information Management as well as chief scientist in data management. She has served as an expert consultant in information security and data management to the Department of Defense, the Department of Treasury, and the Intelligence Community for more than 10 years. Thuraisingham's industry experience includes six years of research and development at Control Data Corporation and Honeywell, Inc. Thuraisingham was educated in the United Kingdom, both at the University of Bristol and the University of Wales.

* * *

Ignacio Aedo (aedo@ia.uc3m.es) holds a degree and a PhD in computer science from Polytechnic University of Madrid and has been actively involved in the domain of interactive systems research since 1990. In this period, he has participated in several national and international projects related to interactive systems. He is author of several books about interactive systems papers published in journals and conferences. His main research interest includes human-computer interaction issues, Web engineering, access control models, and technology in education. He also belongs to the scientific committee of several publications and conferences related to their interest areas.

Walid G. Aref (aref@cs.purdue.edu) is an associate professor of computer science at Purdue University (USA). His research interests are in developing database technologies for emerging applications, for example, spatial, multimedia, genomics, and sensor databases. He is also interested in indexing, data mining, and geographic information systems (GIS). Professor Aref's research has been supported by the NSF, Purdue Research Foundation, CERIAS, Panasonic, and Microsoft Corporation. In 2001, he received the CAREER Award from the National Science Foundation. Professor Aref is on the editorial board of the *VLDB Journal* and is a member of the ACM and the IEEE. For more information, visit *www.cs.purdue.edu/faculty/aref.html*.

Tim Berners-Lee (timbl@w3.org) is the 3Com Founders chair and a senior research scientist at the Laboratory for Computer Science and Artificial Intelligence (CSAIL) at the Massachusetts Institute of Technology (MIT) (USA). He directs the World Wide Web Consortium, founded in 1994. In 1989, he invented the World Wide Web, an Internet-based hypermedia initiative for global information sharing while at CERN, the European Particle Physics Laboratory. He wrote the first Web client and server in 1990. His specifications of URIs, HTTP, and HTML were refined as Web technology spread. He is the author of *Weaving the Web*.

Elisa Bertino (bertino@cerias.purdue.edu) is a professor of computer science and electrical and computer engineering at Purdue University (USA) and research director at CERIAS. She is also a professor of computer science at the University of Milano (Italy). She has been a visiting researcher at the IBM Research Laboratory, the Microelectronics and Computer Technology Corporation, Purdue University. Her main research interests include security, database systems, object-oriented technology, and multimedia systems. She is a member of the editorial boards of several scientific journals, including *International Journal of Information Security, ACM Transactions on Information and Systems Security, IEEE Internet Computing,* and *IEEE Transactions on Secure and Dependable Computing*. She is a fellow member of IEEE and of ACM.

Shu-Ching Chen (chens@cs.fiu.edu) received a PhD from the School of Electrical and Computer Engineering at Purdue University (USA) in December 1998. He also received MS degrees in computer science, electrical engineering, and civil engineering from Purdue University. He has been an associate professor in the School of Computer Science (SCS), Florida International University (FIU) (USA) since August 2004. Prior to that, he was an assistant professor in SCS at FIU from August 1999. His research interests include distributed multimedia database systems, security, data mining, and databases. He received an outstanding faculty research award from FIU (2004) and SCS (2002), and outstanding faculty service award from SCS (2004).

Dan Connolly (connolly@w3.org) is a research scientist at the Computer Science and AI Laboratory and a member of the Decentralized Information Group at Massachusetts Institute of Technology (MIT) (USA). He is also the activity lead for the URI/addressing activity at the World Wide Web Consortium. He has been involved in the standardization of HTML, XML Schema, RDF, and OWL among a number of other W3C activities.

Paloma Dìaz (pdp@inf.uc3m.es) has a degree and a PhD in computer science from the Polytechnic University of Madrid. She is a full professor at the University Carlos III of Madrid (Spain), where she is the head of the DEI Laboratory (dei.inf.uc3m.es). Her research interests mainly concern topics such as hypermedia and Web engineering, software development methodologies, and formal models for representing, reusing, and interchanging information and the application of information and communication technologies in education.

A. Durresi (durresi@csc.lsu.edu) is an assistant professor in the Computer Science Department at Louisiana State University (USA). His main research interests are TCP/IP, quality of service, and routing protocols.

Csilla Farkas (farkas@cse.sc.edu) is an assistant professor in the Department of Computer Science and Engineering and director of the Information Security Laboratory at the University of South Carolina (USA). Dr. Farkas' research interests include information security, data inference problem, economic and legal analysis of cyber crime, and security and privacy on the Semantic Web. She is a recipient of the National Science Foundation Career award. The topic of her award is "Semantic Web: Interoperation vs. Security—A New Paradigm of Confidentiality Threats." Dr. Farkas actively participates in international scientific communities as program committee member and reviewer. Her e-mail address is farkas@cse.sc.edu. For more information, visit *http://www.cse.sc.edu/~farkas.*

Eduardo Fernández-Medina Patón (eduardo.fdezmedina@uclm.es) has a PhD and MSc in computer science. He is an assistant professor at the Escuela Superior de Informática of the Universidad de Castilla-La Mancha at Ciudad Real (Spain). His research activity is security in databases, datawarehouses, Web services, and information systems, and also in security metrics. He is co-editor of several books and chapter books on these subjects and has several dozens of papers in national and international conferences. He participates at the ALARCOS research group of the Department of Computer Science at the University of Castilla-La Mancha in Ciudad Real, Spain. He belongs to various professional and research associations (ATI, AEC, AENOR, IFIP WG11.3, etc.).

Félix J. García Clemente (fgarcia@dif.um.es) is a lecturer in the Department of Computer Engineering of the University of Murcia (Spain). His research interests include network security, security infrastructures, security policy architectures, and IPv6. He received an MSc in computer engineering from the University of Murcia.

Arif Ghafoor (ghafoor@ecn.purdue.edu) is currently a professor in the School of Electrical and Computer Engineering at Purdue University (USA), and is the director of the Distributed Multimedia Systems Laboratory and Information Infrastructure Security Research Laboratory. He has been actively engaged in research areas related to database security, parallel and distributed computing, and multimedia information systems. Dr. Ghafoor has served on the editorial boards of various journals. He has co-edited a book entitled *Multi-*

media Document Systems in Perspectives and has co-authored a book titled *Semantic Models for Multimedia Database Searching and Browsing* (Kluwer Academic Publishers, 2000). Dr. Ghafoor is a fellow of the IEEE. He has received the IEEE Computer Society 2000 Technical Achievement Award for his research contributions in the area of multimedia systems.

Antonio F. Gómez Skarmeta (skarmeta@dif.um.es) is an assistant professor in the Department of Information and Communications Engineering of the University of Murcia (Spain). He received an MSc in computer science from the University of Granada and a PhD in computer science from the University of Murcia. His research interests include distributed systems and security and mobility in computer networks.

Carlos A. Gutiérrez García (carlos.gutierrez@stl.es) has an MSc from the Autonomous University of Madrid (Spain) and is currently a PhD candidate at the University of Castilla—La Mancha. He has developed his professional activity in national and international companies making consultancy activities and, at the moment, he is Internet analyst at Sistemas Técnicos de Loterías del Estado (State Lotteries' Technical Systems) (Spain). His research activity is focused on Web services security and secure software architectures. He has several papers in international conferences, and he has published diverse articles in national and international magazines on these subjects. He is participating at the ALARCOS research group and is an ACM member.

Dilek Hakkani-Tür (dtur@research.att.com) is a senior technical staff member in the Voice Enabled Services Research Department at AT&T Labs-Research in Florham Park, New Jersey (Spain). Prior to joining AT&T in 2001, she received her BS from Middle East Technical University (1994) and MSc and PhD degrees from Bilkent University, Department of Computer Engineering (1996 and 2000, respectively). She visited Carnegie Mellon University, Language Technologies Institute in 1997 and Johns Hopkins University, Computer Science Department in 1998. In 1998 and 1999, she visited SRI International, Speech Technology and Research Labs. She co-authored more than 30 papers in natural language and speech processing.

Jim Hendler (hendler@cs.umd.edu) is a professor at the University of Maryland (USA) and co-director of the Maryland Information and Network

Dynamics Laboratory. Hendler was the recipient of a 1995 Fulbright Foundation Fellowship, is a former member of the US Air Force Science Advisory Board, and is a Fellow of the American Association for Artificial Intelligence. He is also the former chief scientist of the Information Systems Office at the US Defence Advanced Research Projects Agency (DARPA) where he created the DARPA Agent Markup Language Initiative, is the editor-in-chief of *IEEE Intelligent Systems*, and is on the board of reviewing editors for *Science*.

S. Sitharama Iyengar (iyengar@bit.csc.lsu.edu) is chairman and Roy Paul Daniels chaired professor of computer science at Louisiana State University (USA). He is also Satish Dhawan chaired professor at the Indian Institute of Science. He has been awarded the Distinguished Alumnus Award at the Indian Institute of Science in March 2003. He has been involved with research in high-performance algorithms, data structures, sensor fusion, data mining, and intelligent systems since receiving his PhD in 1974 at Mississippi State University and his MS from the Indian Institute of Science in 1970. His publications include 13 books (authored or co-authored textbooks; Prentice-Hall, CRC Press, IEEE Computer Society Press, John Wiley & Sons, etc.) and more than 280 research papers in refereed journals and conferences. He has served as an associate editor for IEEE and as guest editor for the *IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on SMC, IEEE Transactions on Software Engineering, Journal of Theoretical Computer Science, Journal of Computer and Electrical Engineering*, and so on. He is also a fellow of the Association of Computing Machinery (ACM), a fellow of the IEEE, and a fellow of the American Association of Advancement of Science (AAAS).

James B.D. Joshi (jjoshi@mail.sis.pitt.edu) is an assistant professor in the Department of Information Sciences and Telecommunications at the University of Pittsburgh (USA). He is a coordinator of the Laboratory of Education and Research in Security Assured Information Systems (LERSAIS) at the University of Pittsburgh. He received an MS in computer science (1998) and a PhD in computer engineering (2003) from Purdue University. His research interests are information systems security, database security, distributed systems, and multimedia systems. He is a member of the ACM and the IEEE.

Rajgopal Kannan (rkannan@bit.csc.lsu.edu) obtained a BTech in computer science and engineering from IIT-Bombay (1991) and a PhD in computer science from the University of Denver (1996). He is currently an assistant professor in the Computer Science Department at Louisiana State University (USA). His areas of interest are in wireless sensor networks, data security, optical networks, routing and multi-casting protocols, distributed systems, algorithms, and game-theory.

Latifur Khan (lkhan@utdallas.edu) has been an assistant professor of the Computer Science Department at the University of Texas at Dallas (USA) since September 2000. He received his PhD and MS in computer science from the University of Southern California (USC) (August 2000 and December 1996, respectively). Professor Khan is currently supported by grants from the *National Science Foundation (NSF), Alcatel, USA*, and has been awarded the *Sun Equipment Grant*. Dr. Khan has more than 50 articles, book chapters, and conference papers focusing in the areas of database systems, multimedia information management, data mining in bio-informatics, and intrusion detection. Professor Khan has also served as a referee for database journals, conferences (e.g., IEEE TKDE, KAIS, ADL, VLDB), and he is currently serving as a program committee member for the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD2005), ACM 14th Conference on Information and Knowledge Management (CIKM 2005), International Conference on Database and Expert Systems Applications DEXA 2005, and International Conference on Cooperative Information Systems (CoopIS 2005), and program chair of ACM SIGKDD International Workshop on Multimedia Data Mining 2004.

Tsau Young Lin (tylin@cs.sjsu.edu) received his Ph D from Yale University and now is a professor at the Department of Computer Science, San Jose State University (USA). He has served as an editor, associate editor, member of editorial boards on several international journals (e.g., *Data Mining and Knowledge Discovery*), and (co-)chair and member of program committees of conferences. His interests include approximate retrievals, data mining, data warehouse, data security, and novel computing methodologies (granular, rough, and soft computing). He has received the best contribution award in ICDM 2001. He is the founding president of International Rough Set Society, the chair of the task force on granular computing in IEEE—Computational Intelligence Society.

Gregorio Martínez Pérez (gregorio@dif.um.es) is a lecturer in the Department of Information and Communications Engineering of the University of Murcia (Spain). His research interests include security and management of IPv4/IPv6 communication networks. He received an MSc and PhD in computer engineering from the University of Murcia.

Susana Montero (smontero@inf.uc3m.es) has a degree in computer science from the Universidad Carlos III of Madrid (Spain). Since in 1999, she has been a lecturer in the Department of Computer Science at the same university. Her research interests include hypermedia development methodologies, CASE, and knowledge.

Stanley R.M. Oliveira (stanley@cnptia.embrapa.br) obtained his MSc in computer science from the Federal University of Paraiba, Brazil (1995) and his PhD in computer science from the University of Alberta, Canada (2004). Between 1989 and 1995, he was a system analyst with the Brazilian Agricultural Research Corporation—Embrapa (Brazil). Since June 1995, he has worked as a researcher at Embrapa Information Technology in Campinas, SP, Brazil. He has published several papers, most of them in international conferences. His main research interests include privacy-preserving data mining, database security, information hiding, and modeling and simulation.

Andrea Perego (perego@dico.unimi.it) is a PhD student at the Department of Computer Science and Communication of the University of Milan (Italy). He holds an undergraduate degree in humanities and an MS in computer and communication sciences. His main research interests concern the Semantic Web and Internet filtering and rating systems, with special emphasis on the development of multi-strategy filtering systems, supporting different types of metadata structures. He has been involved in the EU project EUFORBIA.

Mario Piattini Velthius (mpiattin@inf-cr.uclm.es) is an MSc and PhD holder in computer science by the Politechnical University of Madrid, certified information system auditor by ISACA (Information System Audit and Control Association), associate professor at the Escuela Superior de Informática of the Castilla—La Mancha University (Spain), author of several books and papers on databases, software engineering, and information systems. He leads the ALARCOS research group of the Department of Computer Science at the

University of Castilla—La Mancha in Ciudad Real, Spain. His research interests are advanced database design, database quality, software metrics, object-oriented metrics, and software maintenance.

Daniel Sanz (dsanz@inf.uc3m.es) has a degree in computer science from the Universidad Carlos III of Madrid (Spain). Starting in 2002, he has been a lecturer in the Department of Computer Science at the same university. His research interests include hypermedia development methodologies, access control models, Web services, and their applications to Web engineering.

Yücel Saygin (ysaygin@sabanciuniv.edu) received his PhD from the Department of Computer Engineering at Bilkent University, Turkey (2001). During his PhD studies, he worked as a visiting researcher at Purdue University, Indiana Center for Database Systems (February 1999 to July 2000). His main research interests include data mining and privacy issues in data management. He published his research results in international journals like *IEEE Transactions on Fuzzy Systems*, *IEEE Transactions on Knowledge and Data Engineering*, and *IEEE Transactions on Engineering Management, Information Sciences, Information Systems*. He is currently an assistant professor at Sabancý University, Faculty of Engineering and Natural Sciences (Turkey).

Mei-Ling Shyu (shyu@miami.edu) has been an assistant professor at the Department of Electrical and Computer Engineering, University of Miami (USA) since January 2000. She received her PhD from the School of Electrical and Computer Engineering, Purdue University (USA) (1999). She also received her MS in computer science, MS in electrical engineering, and MS in restaurant, hotel, institutional, and tourism management from Purdue University (1992, 1995, and 1997, respectively). Her research interests include data mining, security, multimedia networking, multimedia database systems, multimedia information systems, and database systems.

Konstantina Stoupa (kstoupa@acn.gr) received her BSc in computer science from Aristotle University of Thessaloniki, Greece and an MS in business administration from the University of Macedonia of Thessaloniki (Greece). She is currently a PhD student in the Department of Informatics in Aristotle University of Thessaloniki. Since 2002, she has been collaborating with the Information Management Department of the Technological Educational Institute of Kavala, Greece. Her primary research interests include access control systems and delegation of rights and authorizations using XML.

Gökhan Tür (gtur@research.att.com) received his BS, MS, and PhD from the Department of Computer Science, Bilkent University, Turkey (1994, 1996, and 2000, respectively). Between 1997 and 1999, he visited the Center for Machine Translation of CMU, then the Department of Computer Science of Johns Hopkins University, and then the Speech Technology and Research Lab of SRI International. Dr. Tür is a senior technical specialist at AT&T Labs. His research interests include speech and language processing, machine learning, and information retrieval and extraction. His work has been published in several refereed journals and presented in more than 30 international conferences.

Athena Vakali (avakali@csd.auth.gr) received a BSc in mathematics from the Aristotle University of Thessaloniki (Greece), an MSc in computer science from Purdue University, USA (with a Fulbright scholarship), and a PhD in computer science from the Department of Informatics at the Aristotle University of Thessaloniki. Since 1997, she has been a faculty member of the Department of Informatics, Aristotle University of Thessaloniki, Greece (currently she is an assistant professor). Her research interests include design, performance, and analysis of storage subsystems and data placement schemes for multimedia and Web-based information. She is working on Web data management, and she has focused on XML data storage issues. She has published several papers in international journals and conferences. Her research interests include storage subsystem's performance, XML and multimedia data, and management and data placement schemes.

Daniel J. Weitzner (djw@w3.org) is director of the World Wide Web Consortium's Technology and Society activities. He also holds an appointment as principle research scientist at Massachusetts Institute of Technology's (MIT) Computer Science and Artificial Intelligence Laboratory and teaches Internet public policy at MIT. Before joining the W3C, Mr. Weitzner was co-founder and deputy director of the Center for Democracy and Technology, an Internet civil liberties organization in Washington, DC, and was also deputy policy director of the Electronic Frontier Foundation.

Stephen S. Yau (yau@asu.edu) is currently a professor in the Department of Computer Science and Engineering at Arizona State University, Tempe (USA). He served as the chair of the department from 1994 to 2001. He was previously with the University of Florida, Gainesville and Northwestern University, Evanston, Illinois. He served as the president of the Computer Society of the Institute of Electrical and Electronics Engineers (IEEE), and the editor-in-chief of *IEEE Computer* magazine. His current research is in software engineering, distributed computing systems, adaptive middleware, and trustworthy computing. He received a BS from National Taiwan University, Taipei, and MS and PhD from the University of Illinois, Urbana, all in electrical engineering. He is a fellow of the IEEE and American Association for the Advancement of Science. He received the Louis E. Levy Medal of the Franklin Institute and the Tsutomu Kanai Award of the IEEE Computer Society. For more information, visit *http://www.eas.asu.edu/~seg/yau.*

Osmar R. Zaïane (zaiane@cs.ualberta.ca) is an associate professor in computing science at the University of Alberta (Canada), obtained an MS in electronics (Paris, 1989) and an MS in computer science (Quebec, 1992). He obtained his PhD from Simon Fraser University, Canada (1999) under the supervision of Dr. Jiawei Han. He has research interests in novel data mining algorithms; Web, text, and image mining; and information retrieval. He has published more than 60 papers in international conferences and journals as well as multiple book chapters on multimedia and Web mining. He was the co-chair of the ACM SIGKDD Workshop on Multimedia Data-Mining in 2000, 2001, and 2002 as well as the ACM SIGKDD WebKDD Workshop in 2002 and 2003. He is the *ACM SIGKDD Explorations* associate editor.

Index

Q

R

S