FROM XML TO RDF: SYNTAX, SEMANTICS, SECURITY, AND INTEGRITY (INVITED PAPER)

C. Farkas, V. Gowadia, A. Jain, and D. Roy
Information Security Lab
Department of Computer Science and Engineering
University of South Carolina
Columbia, SC 29208

Abstract   In this paper we evaluate security methods for eXtensible Markup Language (XML) and the Resource Description Framework (RDF). We argue that existing models are insufficient to provide high assurance security for future Web-based applications. We begin with a brief overview of XML access control models, where the protection objects are identified by the XML syntax. However, these approaches are limited to handle updates and structural modifications of the XML documents. We argue that XML security methods must be based on the intended meaning of XML and the semantics of the application using XML. We identify two promising research directions to extend the XML model with semantics. The first approach incorporates traditional database concepts, like key and integrity constraints, in the XML model. The second approach aims to associate XML documents with metadata supporting Web-based applications. We propose the development of security models based on these semantics-oriented approaches to achieve high assurance. Further, we investigate the security needs of Web metadata, like RDF, RDFS, and OWL. In particular, we study the security risks of unwanted inferences and data aggregation, supported by these languages.

1.   Introduction

The rapid development of the World Wide Web (WWW) has led to the development of machine understandable, self describing syntax to exchange data. Presently, the eXtensible Markup Language (XML) is the most widely used language to support Web-based applications. To further facilitate these applications, the Semantic Web community has proposed languages, such as the Resource Description Framework

(RDF), and the Web Ontology Language (OWL), for representation of metadata. In addition to the functional requirements, these future applications must also provide data and application security.

During the last five years several access control models have been developed for XML. However, these models target only the simplest interpretation of XML, its purely syntactic form. While this approach might be suitable for some applications, it is unsatisfactory to support general Web application.

In this paper we propose a different approach for XML security, originating from research aiming to extend XML with semantics. We consider two main research directions to extend the XML model with semantics. The first approach extends the XML model with traditional database concepts, like keys and database constraints. The second approach aims to associate XML documents with metadata, supporting Web-based applications. We believe, that security models must be developed based on these semantics-oriented approaches to achieve high-assurance and flexible security.

We start with an overview of XML access control models developed on top of XML syntax. While these models are sufficient to provide secure read accesses to XML, they are limited to handle updates and document restructuring. We show, that these operations may cause violations of confidentiality, integrity, and availability. We present approaches, that seem promising from the perspective of security, to represent the intended meaning of XML. In particular, we present research that extends the XML syntax with RDF, RDFS, and OWL metadata. These metadata facilitate XML restructuring, XML data integration, identification of syntactically different but semantically equivalent XML documents, and to identify security objects.

Although the number of research and development efforts to provide semantics aware security for Web technologies and applications is increasing, these works only target a small fraction of the necessary research. Future work, based on precise formulation of data and application semantics, need to be done. We propose future research directions, including representation of data and applications semantics, development of security models based on these semantics, analysis of the need for metadata security, and study of the inference and aggregation problem in semantic data.

The organization of the paper is as follows. In Section 2 we give an overview of the XML access control research, and its limitations. Section 3 describes research extending the XML model with semantics and developing security models based on these semantics. Section 4

contains our initial evaluation and results on securing metadata. Finally, we conclude and recommend future research directions in Section 5.

2. Extensible Markup Language

The simplest interpretation of an XML document is a tree-structure, composed of properly nested element nodes. In textual representation of XML documents each subtree is delimited by a pair of start and end tags of element name. Each element has zero or more child nodes, which may include other element nodes, text nodes, and attribute nodes. Cardinality constraints and special attributes, like id and idref, allow to express some restrictions on the XML tree.

XML is being increasingly used to support Web-based applications. In addition to the application specific requirements, these applications also require data integrity, confidentiality, and availability. Authorization models, based on syntactic XML trees, identify protection objects as subtrees (collection of nodes) of the XML trees. In this section we give an overview of the existing (syntax-based) XML access control models, point out limitations of these models, and argue that access control should be defined on the intended meaning of XML formatted data, rather than the presentation syntax.

2.1 XML Security

During the last five years, several discretionary access control models [7, 14, 26] have been developed for XML trees. Protection objects correspond to XML nodes, identified by path expressions. These models support authorization propagation, conflict resolution, and expression of obligation and provision at varying degree. They may also support schema-level (i.e., DTD or XML Schema) or data-level (i.e., XML instance) specification of security policies.

The XML Access control model developed by Bertino et al. [8, 9, 6] provides flexible security granularity and considers the case when XML documents do not conform to a predefined Document Type Definition (DTD). The proposed access control model can be used for DTD-based and document-based policies. Security objects are specified by a path expression, identifying one or more nodes in the XML document or DTD. Propagation rules determining access control restrictions for the descendant nodes are also supported.

Damiani et al. [13, 14] defines and enforces access restrictions on the XML document structure and content. The authors propose the construction of partial views of the XML documents, such that the views sat-

isfy the security requirements. Security objects are specified by XPath expressions identifying element or attribute nodes or their collections.

The models proposed by Bertino et al. and Damiani et al. reach a binary decision for granting or denying access to the nodes identified by path expressions. Kudo et al. [25, 26] propose an access control model that provides provisional authorizations [22]. Provisional access control allows to express additional requirements that users must satisfy if their accesses are permitted.

Murata et al. [29] introduce a static analysis technique based on string automata to reduce the overhead of runtime security checking. Given an access control policy, a query expression, and an optional schema, static analysis determines if the query potentially violates the security policy. Static analysis can be performed without evaluating any query expression against an actual database. Run-time (i.e., data level) checking is required only when static analysis is unable to determine whether to grant or deny access requests.

Gowadia and Farkas [18] present an RDF-based access control framework to support context based access control. RDF statements are used to represent meta-data, including security objects and policies. Their aim is to increase data availability while providing security. In [19] the authors address efficient enforcement of their model by using bottom-up tree automata to represent security objects. They support both data and schema level evaluation.

2.2   Limitations of Syntax-Based XML Security Models

Correct enforcement of existing access control models requires that the structure of the document does not change and that the security classifications of the nodes increase downward in the XML tree. Changes in the XML structure or data may result in incomprehensible security policy or data loss. In this section we present two such examples. The first example shows the limitation of handling updates in multilevel secure XML documents. The second example shows the problem of structural rewriting of XML documents.

XML Updates: The focus of access control models developed so far has been on providing read access to the users, without fully considering write access. For instance, when a delete operations is issued, the entire subtree of the deleted nodes is removed [12]. This means that users may delete nodes they are not authorized to read if these nodes are in the subtree of an authorized node. Such blind deletes lead to loss of important information.
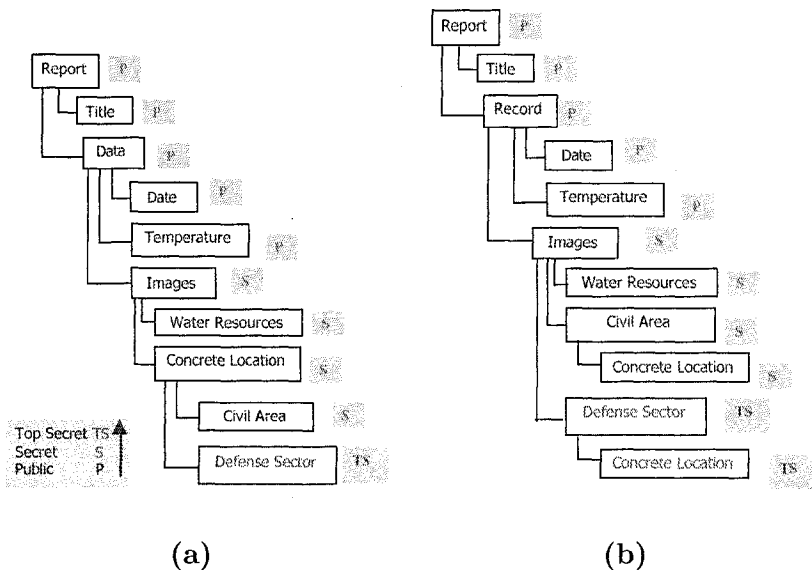
(a)   (b)

Figure 1.   Example MLS XML documents

To illustrate this problem consider the XML document shown in Figure 1(a). The document contains data received from satellite images. The data is classified at three security levels: $TopSecret > Secret > Public$. If a user with Public clearance deletes the <Data> element, all subtrees of <Data> are also deleted. This includes the Public level <Date> and <Temperature> as well as the Secret level <Images> and its subtrees. This will -incorrectly- reduce the data availability for Secret and Top Secret users.

Finding a secure and correct solution to handle delete is not trivial. Other approaches include:

■ Delete only the read authorized (viewable) nodes and allow fragmentation of the XML tree. But then, future querying and policy enforcement will be limited. If the dangling subtrees are connected to the nearest parent nodes then the XML schema would also be violated.

■ Refuse to delete any node that has nodes in its subtree that are not authorized for the user. However, this solution would create a covert channel.

Clearly none of the above solutions is acceptable. Further work is needed to evaluate updates in XML documents with different security requirements for the nodes.

Restructuring XML Documents: Another problem with syntax-based access control models is, that it is not possible to have a single access control policy for different XML structures even if they contain the same data. For example, a syntactic policy for XML document in Figure 1(a) cannot be used for securing the XML document shown in Figure 1(b) even if they have the same data values. Observation of the two XML trees show that they only differ in their structure and the tag name <Data> in Figure 1(a) is replaced with tag name <Record> in Figure 1(b).

Such structural variations often arise during merger of two or more organizations, because each organization may already have its own XML data, stored according to local schemas. After merger the organizations still need to enforce local access control policies over the combined data. To ensure correct enforcement, it is necessary to provide conflict resolution strategies and transformation of policies between the different syntactic forms. Performing these transformations by humans is time consuming and may lead to errors. Development of automated tools require that the intended semantics of the XML formatted data is represented in a machine-understandable format.

3.   XML and Semantics

Our belief is that security models must be based on semantics rather than syntax. Lack of capabilities to handle data semantics will result in inflexible policies that cannot handle application specific requirements.

Several researchers addressed the problem of extending the current XML model with semantics. We study two of these approaches from the perspective of XML security: 1) database oriented, to support expressiveness required by databases, and 2) Web Services oriented, to support application specific semantics. This section gives an overview of these approaches.

3.1   XML as database

Database researchers attempt to extend the XML model to support database semantics in XML. Although DTDs and XML Schema allow simple constraints for XML, these type of constraints are not sufficient for constraints usually present in databases. Buneman, Davidson, and Fan [10, 16, 15] develop key and integrity constraints for XML. Key con-

straints are especially important to express semantics of objects identities, thus necessary to identify security protection objects.

Considering XML from the database perspective also led to the development of query languages, like XQuery, XML-QL. The need for efficient query processing led to formal data models and query optimization. Jagadish et al. [21] present a Tree Algebra for XML queries (TAX). TAX is an extension of relational algebra and can express most XML query operations. Hung et al. [20] propose TOSS, an extension of TAX with the semantics of terms stored in TAX databases. The authors incorporate a similarity enhances ontology (SEO) to allow queries over syntactically different by "similar" terms.

Liu et al. [27] propose an XML Semantics Definition Language (XSDL) to express XML author's intended meaning. In XSDL, XML semantics is defined in terms of OWL DL ontology. The mapping between the XML and the ontology is provided using Schema Adjuncts Framework (SAF).

Unfortunately, with the exception of some initial attempts, none of the XML security models incorporate these semantics-aware approaches. We believe that these approaches would be useful to overcome the limitations of current security models. Therefore, we recommend further research to evaluate their applicability for security. The following section will give an overview of the existing, semantic-aware access control models for XML and XML-like languages.

3.2   XML Security and Semantics

Stoica and Farkas [17, 35, 36] propose a method similar to Liu et al. [27]. They manipulate XML documents according to metadata associated to them.

Secure XML Views: In [35] Stoica and Farkas address the restriction that security classifications of the nodes must increase downwards in an XML tree. The authors propose techniques for generating secure and semantic-conflict free XML views from a multilevel secure (MLS) XML. They propose the use of two graphs, a Minimum Semantic Conflict Graph (MSCG) and a Multi-Plane DTD Graph (MPG). MSCG contains all semantic relationships among the XML tags that must be preserved within any partial view. MPG captures the structural relationships among tags and their security classifications.

XML Correlation with Ontologies: In [17] and [36] Stoica and Farkas show that large collections of distributed XML documents are exposed to inference attacks through data correlation and replication. They propose that XML documents to be mapped to ontologies (Fig-
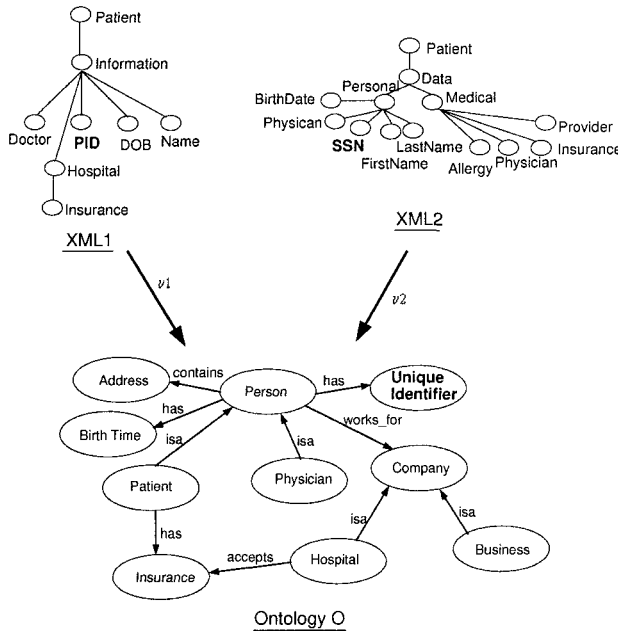
Figure 2.   XML mapping to Ontology

ure 2) to convey intended meaning. This mapping is used to identify semantically equivalent XML nodes. Detection of replicated XML data and association among (distributed) XML nodes is aided by generalization of XML terms based on the ontology. For example, the Correlated Inference Procedure detects correlated information under different security classification and syntactic format.

Concept level Access Control: Qin and Atluri [30] propose an access control model to define authorizations on the ontological concepts linked to the semantically annotated Web pages. The access control policies are defined on concept and enforced on the data instances.

XML Updates: Roy [32] addresses the problem of secure and integrity preserving deletes in MLS XML documents. She suggests the use of a unique new domain to relabel nodes that are deleted by a user but the delete would result in document fragmentation or data loss. For example, in Section 2.2 we showed that the deletion of <Data> node would result in disconnecting <Images> from the root. The proposed solution would remove such "deleted" nodes from the view of Public users by relabeling them with the {Deleted} domain. However, these relabeled nodes would still be visible to Secret and Top Secret users, with the indication that they were deleted by a Public user. Clearances
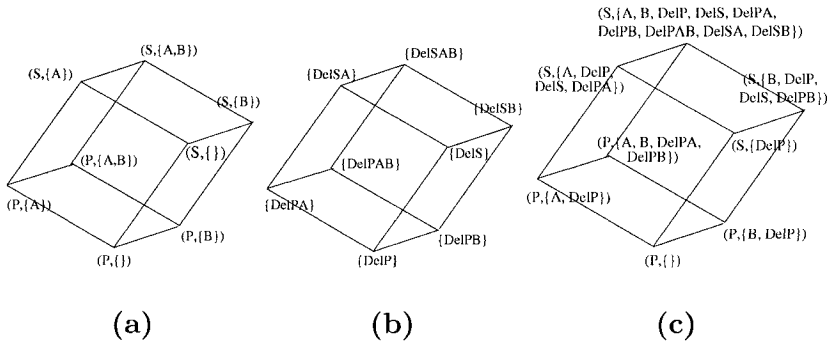
Figure 3.   Lattice Structure of Security Levels

of the subjects are assigned such that they can access deleted nodes with strictly dominated labels. Figure 3 shows the original and part of the modified security lattice.

However, work by Roy does not fully solve the problems of XML updates. For example, it preserves minimum nodes required to preserve document structure but does not consider semantics of high-level security data. Can we still use a Top Secret image after its date and location have been deleted? What is the meaning of a Secret value that is calculated from "deleted" Public values? We believe that ontologies play a crucial role to develop semantically correct and secure solutions for the above problems. For example, they would supply data semantics, similar to referential integrity in relational databases. An other area is to evaluate the key constraints in XML documents and their effect on security. Is there a polyinstantiation problem is XML?

XML Normal Form: Finally, machine understandable representation of the intended XML semantics need to be addressed. We propose an approach using ontologies to form equivalence classes of syntactically different but semantically (ontologically) equivalent XML documents. XML Normal Form, a syntactic construct, is used to represent each equivalence class. The proper syntax of XML Normal Form is determined by the ontology representing XML semantics.

Synchronized Multimedia Integration Language Security: Kodali et al. [23, 24] develop security framework for Synchronized Multimedia Integration Language (SMIL) formatted streaming data. SMIL, an XML-like language, supports operational semantics. The authors provide language-based security that respects continuity and synchronization constructs of SMIL. They introduce the concept of SMIL Normal Form, representing the equivalence class of syntactically different but

semantically equivalent SMIL document. They develop models for Discretionary (DAC), Mandatory (MAC), and Role-Based (RBAC)Access Control, and address issues like unbreakability of atomic SMIL units.

3.3   Secure XML for Web Services

Web Services (WS) are the Web based ubiquitous applications built on open standards. WS can be advertised, discovered, and invoked over the Web. They are published on the Web using WSDL (Web Services Description Languages) [11]. UDDI [5] is the registry where they are listed in the directory and can be discovered by the requester service. The interacting Web Services exchange all the data and requests in messages format using SOAP (Simple Object Access Protocol) [28]. All of these standards use XML as the underlying data syntax for data discovery, interchange, and processing. All of these interactions occur at the syntactic level where the services are discovered from UDDI by keyword based search. WS-Security specification [4] uses XML digital signature to sign the SOAP messages, XML encryption to encrypt the messages and data, XACML for access control. In addition to this it uses PKI, Kerberos and other conventional security mechanism to provide secure data interchange and processing.

Currently WS use ontologies to improve the performance of automated discovery of registered services. WS security must be able to handle application and business specific requirements.

4.   Protecting Metadata

One of the main achievement of the envisioned Semantic Web is the use of complex relationships between entities to support interoperation and data integration. These relationships may also lead to entailments of new facts and relationships. Sheth et al. [2, 3, 34] develop inferencing tools that treat sequence of properties as a new type of relationship. These relationships capture connections and similarities between data resources which are not directly connected. The authors give real life application examples of how to identify useful associations in the domains of businesses and national security. For example, the Passenger Identification, Screening, and Threat Analysis application (PISTA) [33] involves discovering and preventing threats for aviation safety. PISTA demonstrates the use of semantic associations in calculating the possible risk from passengers in a given flight. It extracts relevant metadata from different information resources and channels including government watch-lists, commercial data, flight databases, and historical passenger data. PISTA uses semantic-based knowledge discovery techniques
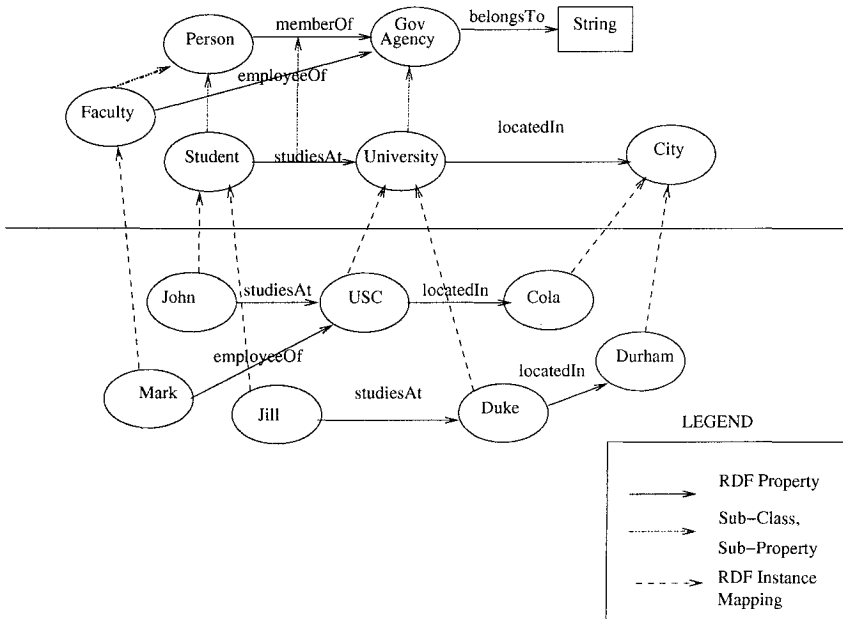
Figure 4.   Example RDF Schema and Instance Data

to identify suspicious patterns and categorize passengers into different groups.

Metadata and Security: RDF and RDFS have well defined semantics and entailment capabilities. While these capabilities are needed to improve data integration and interoperation, they may also be used to disclose sensitive data or to disclose a sensitive pattern. Access control models for RDF and RDFS must consider these inferencing capabilities.

Although some of the XML security models utilize metadata to enhance the security, they do not develop security models for metadata. However, the amount of metadata, stored in RDF, RDFS, and OWL format, is increasing; methods and tools are being developed to store, manipulate, and query metadata [1]. Making these metadata publicly available, i.e., for Web applications, raises new security and privacy concerns. Can we use conventional security models, developed for XML and RDBMS, to protect Web data and corresponding metadata? Before answering this question, we need to evaluate the inferencing capabilities of RDF, RDFS, and OWL. For example, RDF and RDFS entailments may generate assertions that are not explicitly stored but could be inferred. From the security perspective this new data should also be secured by the authorization framework and should not violate the security policy.

Jain and Farkas (http://www.cse.sc.edu/research/isl) develop formalism for RDF access control, incorporating RDF and RDF Schema (RDFS) entailments. Security violations occur if a sensitive statement can be entailed from non-sensitive statements. RDF protection objects are represented as RDF-patterns (triples) along with the corresponding security labels. The model has flexible security granularity that allows expressing restrictions on a single resource, property, value, or any combination of these. Conflict resolution strategy addresses the problem of inconsistent classification. The authors also develop techniques to assign security classification to newly generated statements.

For example, consider Figure 4. Assume that the information that USC is a type of GovAgency is confidential. However, releasing the information that <USC rdf:type University> and <University rdfs:subClassOf GovAgency> entails the relationship <USC rdf:type GovAgency>. Even this simple example shows that security models that address entailments must be developed.

A different approach for RDF security is presented by Finin et al. [31]. They propose a policy based access control model for RDF data in an RDF store. The model provides control over the different action modes possible on the RDF store, like inserting a triple, deleting a triple, and querying whether a triple is in the store. The authors define a set of policy rules, enforced by a policy engine to reach the authorization decisions.

We believe that ontologies are crucial for future Semantic Web technologies, providing the basis for representing, acquiring, and utilizing knowledge. Researchers and developers need to consider security aspects of these new technologies and develop appropriate authorization frameworks.

5.   Conclusions

This paper presents a brief overview of XML and RDF data and their security models. Our main aim is to indicate the need of precise formulation of data and application semantics and their use to develop security models. We present initial research results aiming to extend the XML paradigm with formal semantics. We give motivating examples and suggest further research directions.

Also, we believe that RDF and ontology languages play a significant role in developing the Semantic Web. However, only a few of the researchers address the need to develop authorization frameworks for metadata. Methods, capable of handling entailments and complex relationships need to be developed. Further, assurance of the security

methods need to be established. Finally, enforcement and scalability issues need to be studied to achieve practical solutions. This is especially important when considering the open and dynamic nature of the Semantic Web.

6.   Acknowledgment

References

[1] Kowari-metastore. http://www.kowari.org.

[2] B. Aleman-Meza, C. Halaschek, J. B. Arpinar, and A. Sheth. Context-aware semantic association ranking. In *Proceedings of the First International Workshop on Semantic Web and Databases*, pages 33–50. LSDIS Lab, University of Georgia, 2003.

[3] K. Anyanwu and A. Sheth. p-Queries: Enabling Querying for Semantic Associations on the Semantic Web. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 690–699, New York, NY, USA, 2003. ACM Press.

[4] B. Atkinson, G. Della-Libera, S. Hada, and M. Hondo. Web Services Security (WS-Security). http://www-106.ibm.com/developerworks/webservices/library/ws-secure/, April 2002.

[5] T. Bellwood, L. Clment, and C. von Riegen. Universal Description, Discovery and Integration (UDDI) V3.0. http://uddi.org/pubs/uddi-v3.0.1-20031014.pdf, October 2003. OASIS Specification.

[6] E. Bertino, M. Braun, S. Castano, E. Ferrari, and M. Mesiti. Author-X: A Java-based System for XML Data Protection. In *Proc. IFIP WG11.3 Working Conference on Database Security*, The Netherlands, August 2000.

[7] E. Bertino, S. Castano, and E. Ferrari. Securing XML Documents with Author-X. *IEEE Internet Computing*, 5(3):21–31, 2001.

[8] E. Bertino, S. Castano, E. Ferrari, and M. Mesiti. Controlled Access and Dissemination of XML Documents. In *Proc. of 2nd ACM Workshop on Web Information and Data Management*, pages 22–27, Kansas City, 1999.

[9] E. Bertino, S. Castano, E. Ferrari, and M. Mesiti. Specifying and enforcing access control policies for XML document sources. *World Wide Web*, 3(3):139–151, 2000.

[10] P. Buneman, S. Davidson, W. Fan, C. Hara, and W.-C. Tan. Reasoning about keys for XML. *Information Systems*, 28(8):1037–1063, 2003.

[11] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. Web Services Description Language (WSDL) 1.1. http://www.w3.org/TR/wsdl, March 2001.

[12] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. A fine-grained Access Control System for XML documents. *ACM Trans. Inf. Syst. Secur.*, 5(2):169–202, 2002.

[13] E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati. Design and Implementation of an Access Control Processor for XML Documents. In *9th World Wide Web Conference*, The Netherlands, 2000.

[14] E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati. Securing XML Documents. In *Conference on Extending Database Technology*, Prague, March 2002.

[15] W. Fan and L. Libkin. On XML integrity constraints in the presence of DTDs. *J. ACM*, 49(3):368–406, 2002.

[16] W. Fan and J. Simeon. Integrity Constraints for XML. In *Symposium on Principles of Database Systems*, pages 23–34, 2000.

[17] C. Farkas and A. Stoica. Correlated Data Inference in Ontology Guided XML Security Engine. In *Proc. of IFIP WG 11.3 Working Group Conference on Data and Application Security*, 2003.

[18] V. Gowadia and C. Farkas. RDF metadata for XML Access Control. In *Proceedings of the 2003 ACM workshop on XML security*, pages 39–48. ACM Press, 2003.

[19] V. Gowadia and C. Farkas. Tree automata for Schema-level Filtering of XML Associations. *Journal of Research and Practice in Information Technology*, page In Press, 2005.

[20] E. Hung, Y. Deng, and V. S. Subrahmanian. TOSS: an extension of TAX with Ontologies and similarity queries. In *SIGMOD '04: Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 719–730, New York, NY, USA, 2004. ACM Press.

[21] H. V. Jagadish, L. V. S. Lakshmanan, D. Srivastava, and K. Thompson. TAX: A Tree Algebra for XML. In *Proceedings of DBPL'01*, pages 149–164, 2001.

[22] S. Jajodia, M. Kudo, and V. S. Subrahmanian. Provisional Authorizations. In *Proc. 1st Workshop on Security and Privacy in E-Commerce*, 2000.

[23] N. Kodali, C. Farkas, and D. Wijesekera. An Authorization Model for Multimedia Digital Libraries. *Journal of Digital Libraries*, 4:139–155, 2004.

[24] N. Kodali, C. Farkas, and D. Wijesekera. Enforcing Semantics Aware Security in Multimedia Surveillance. *Journal on Data Semantics (Springer LNCS)(Invited)*, 2:199–221, 2005.

[25] M. Kudo and S. Hada. XML document security based on provisional authorization. In *CCS '00: Proceedings of the 7th ACM conference on Computer and communications security*, pages 87–96, New York, NY, USA, 2000. ACM Press.

[26] M. Kudo and S. Hada. Access Control Model with Provisional Actions. In *IEICE Trans. Fundamentals*, 2001.

[27] S. Liu, J. Mei, A. Yue, , and Z. Lin. XSDL: Making XML Semantics Explicit. In *Proc. of Semantic Web and Databases, Second International Workshop*, pages 64–83, Toronto, Canada, August 2004.

[28] N. Mitra. SOAP Version 1.2 Part 0: Primer. http://www.w3.org/TR/2003/REC-soap12-part0-20030624/, June 2003.

[29] M. Murata, A. Tozawa, M. Kudo, and S. Hada. XML Access Control using Static Analysis. In *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*, pages 73–84. ACM Press, 2003.

[30] L. Qin and V. Atluri. Concept-level Access Control for the Semantic Web. In *Proceedings of the 2003 ACM workshop on XML security*, pages 94–103. ACM Press, 2003.

[31] P. Reddivari, T. Finin, and A. Joshi. Policy based Access Control for a RDF Store. In *Proceedings of the Policy Management for the Web Workshop*, A WWW 2005 Workshop, pages 78–83. W3C, May 2005.

[32] D. Roy. Multilevel XML Data Model. Master's thesis, University of South Carolina, Columbia, July 2005.

[33] A. Sheth, B. Aleman-Meza1, I. B. Arpinar, C. Halaschek, C. Ramakrishnan, C. Bertram, Y. Warke, D. Avant, F. S. Arpinar, K. Anyanwu, and K. Kochut. Semantic Association Identification and Knowledge Discovery for National Security Applications. *Special Issue of JOURNAL OF DATABASE MANAGEMENT on Database Technology for Enhancing National Security, Ed. Lina Zhou. (Invited paper).*, August 2003.

[34] A. Sheth, C. Bertram, D. Avant, B. Hammond, K. Kochut, and Y. Warke. Managing semantic content for the web. *IEEE Internet Computing*, 6(4):80–87, 2002.

[35] A. Stoica and C. Farkas. Secure XML Views. In *Proc. of IFIP WG11.3 Working Group Conference on Database and Application Security*, 2002.

[36] A. Stoica and C. Farkas. Ontology guided Security Engine. *Journal of Intelligent Information Systems*, 23:209–223, 2004.