Security and Privacy for Web Databases and Services

Elena Ferrari[1] and Bhavani Thuraisingham[2]

[1] Università dell'Insubria, 22100 Como, Italy
[2] The National Science Foundation, Arlington, VA, USA

Abstract. A semantic web can be thought of as a web that is highly intelligent and sophisticated and one needs little or no human intervention to carry out tasks such as scheduling appointments, coordinating activities, searching for complex documents as well as integrating disparate databases and information systems. While much progress has been made toward developing such an intelligent web, there is still a lot to be done. For example, there is little work on security and privacy for the semantic web. However, before we examine security for the semantic web we need to ensure that its key components, such as web databases and services, are secure. This paper will mainly focus on security and privacy issues for web databases and services. Finally, some directions toward developing a secure semantic web will be provided.

1 Introduction

Recent developments in information systems technologies have resulted in computerizing many applications in various business areas. Data has become a critical resource in many organizations, and, therefore, efficient access to data, sharing the data, extracting information from the data, and making use of the information has become an urgent need. As a result, there have been many efforts on not only 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 (DBMSs), 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 even greater demand for managing data, information, and knowledge effectively. There is now so much data on the web that managing them with conventional tools is becoming almost impossible. As a results, to provide interoperability as well as warehousing between 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 databases, 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

E. Bertino et al. (Eds.): EDBT 2004, LNCS 2992, pp. 17–28, 2004.

it is even more important to protect the data and information as numerous individuals now have access to them. Therefore, we need effective mechanisms for securing data and applications. The web is now evolving into the semantic web. Semantic web is about ensuring that web pages can be read and understood by machines. The major components for the semantic web include web infrastructures, web databases and services, and ontology management and information integration. There has been a lot of work on each of these three areas. However, very little work has been devoted to security. If the semantic web is to be effective, we need to ensure that the information on the web is protected from unauthorized accesses and malicious modifications. We also need to ensure that individual's privacy is maintained. This paper focuses on security and privacy related to one of the component for the semantic web, that is, for web databases and services.

The organization of this paper is as follows. In Section 2 we give some background information on web databases and services. Security and privacy for web databases will be discussed in Section 3, whereas security and privacy for web services will be discussed in Section 4. Some issues on developing a secure semantic web will be discussed in Section 5. The paper is concluded in Section 6.

2 Background on Web Databases and Services

This paper focuses on security and privacy for web databases and services and therefore in this section we provide some background information about them.

2.1 Web Data Management

A major challenge for web data management is coming up with an appropriate data representation scheme. The question is: is there a need for a standard data model? Is it at all possible to develop such a standard? If so, what are the relationships between the standard model and the individual models used by the databases on the web? The significant development for web data modeling came in the latter part of 1996 when the World Wide Web Consortium (W3C) [15] was formed. This group felt that web data modeling was an important area and began addressing the data modeling aspects. Then, sometime around 1997 interest in XML (Extensible Markup Language) began. This was an effort of the W3C. XML is not a data model. It is a metalanguage for representing documents. The idea is that if documents are represented using XML then these documents can be uniformly represented and therefore exchanged on the web. Database management functions for the web include those such as query processing, metadata management, security, and integrity. Querying and browsing are two of the key functions. First of all, an appropriate query language is needed. Since SQL is a popular language, appropriate extensions to SQL may be desired. XML-QL and XQuery [15] are moving in this direction. Query processing involves developing a cost model. Are there special cost models for Internet database management? With respect to browsing operations, the query processing techniques

have to be integrated with techniques for following links. That is, hypermedia technology has to be integrated with database management technology. Transaction management is essential for many applications. There may be new kinds of transactions for web data management. For example, various items may be sold through the Internet. In this case, the item should not be locked immediately when a potential buyer makes a bid. It has to be left open until several bids are received and the item is sold. That is, special transaction models are needed. Appropriate concurrency control and recovery techniques have to be developed for the transaction models. Metadata management is also a major concern. The question is, what is metadata? Metadata describes all of the information pertaining to a data source. This could include the various web sites, the types of users, access control issues, and policies enforced. Where should the metadata be located? Should each participating site maintain its own metadata? Should the metadata be replicated or should there be a centralized metadata repository? Storage management for Internet database access is a complex function. Appropriate index strategies and access methods for handling multimedia data are needed. In addition, due to the large volumes of data, techniques for integrating database management technology with mass storage technology are also needed. Maintaining the integrity of the data is critical. Since the data may originate from multiple sources around the world, it will be difficult to keep tabs on the accuracy of the data. Appropriate data quality maintenance techniques need thus be developed. Other data management functions include integrating heterogeneous databases, managing multimedia data, and mining. Security and privacy is a major challenge. This is one of the main focus areas for this paper and will be discussed in Section 3.

2.2 Web Services

Web services can be defined as an autonomous unit of application logic that provides either some business functionality features or information to other applications through an Internet connection. They are based on a set of XML standards, namely, the Simple Object Access Protocol (SOAP) [15] – to expose the service functionalities, the Web Services Description Language (WSDL) [15] – to provide an XML-based description of the service interface, and the Universal Description, Discovery and Integration (UDDI) [16] – to publish information regarding the web service and thus making this information available to potential clients. UDDI provides an XML-based structured and standard description of web service functionalities, as well as searching facilities to help in finding the provider(s) that better fit the client requirements. More precisely, an UDDI registry is a collection of entry, each of one providing information on a specific web service. Each entry is in turn composed by five main data structures `businessEntity`, `businessService`, `bindingTemplate`, `publisherAssertion`, and `tModel`, which provide different information on the web service. For instance, the `BusinessEntity` data structure provides overall information about the organization providing the web service, whereas the `BusinessService` data structure provides a technical description of the service.

Searching facilities provided by UDDI registries are of two different types, which result in two different types of inquiries that can be submitted to an UDDI registry: *drill-down pattern inquiries* (i.e., get_xxx API functions), which return a whole core data structure (e.g., `businessTemplate`, `businessEntity`), and *browse pattern inquiries* (i.e., find_xxx API functions), which return overview information about the registered data.

As far as architectural aspects are concerned, three are the main entities composing the Web Service Architecture (WSA): the *service provider*, which is the person or organization that provides the web service, the *service requestor*, which is a person or organization that wishes to make use of the services offered by a provider for achieving its business requirements, and the *discovery agency*, which manages UDDI registries. UDDI registries can be implemented according to either a third-party or a two-party architecture, with the main difference that in a two-party architecture there is no distinction between the service provider and the discovery agency, whereas in a third-party architecture the discovery agency and the service provider are two separate entities. It is important to note that today third-party architectures are becoming more and more widely used for any web-based system, due to their scalability and the ease with which they are able to manage large amount of data and large collections of users.

3 Security and Privacy for Web Databases

Security issues for web databases include secure management of structured databases as well as unstructured and semistructured databases, and privacy issues. In the following sections we discuss all these aspects.

3.1 Security for Structured Databases on the Web

A lot of research has been done for developing access control models for Relational and Object-oriented DBMSs [6]. For example, today most of the commercial DBMSs rely on the System R access control model. However, the web introduces new challenges. For instance, a key issue is related to the population accessing web databases which is greater and more dynamic than the one accessing conventional DBMSs. This implies that traditional identity-based mechanisms for performing access control are not enough. Rather a more flexible way of qualifying subjects is needed, for instance based on the notion of role or credential. Next we need to examine the security impact on all of the web data management functions. These include query processing, transaction management, index and storage management, and metadata management. For example, query processing algorithms may need to take into consideration the access control policies. We also need to examine the trust that must be placed in the modules of the query processor. Transaction management algorithms may also need to consider the security policies. For example, the transaction will have to ensure that the integrity as well as security constraints are satisfied. We need to examine the security impact in various indexing and storage strategies. For example, how

do we store the databases on the web that will ease the enforcement of security policies? Metadata includes not only information about the resources, which includes databases and services, it also includes security policies. We need efficient metadata management techniques for the web as well as use metadata to enhance security.

3.2 Security for XML, RDF, and Ontology Databases

As we evolve the web into the semantic web, we need the capability to manage XML and RDF databases. This means that we need to ensure secure access to these databases.

Various research efforts have been reported for securing XML documents and XML databases [11]. Here, we briefly discuss some of the key points. XML documents have graph structures. The main challenge is thus to develop an access control model which exploits this graph structure in the specification of policies and which is able to support a wide spectrum of access granularity levels, ranging from sets of documents, to single documents, to specific portions within a document, as well as the possibility of specifying both content-dependent and content-independent access control policies. A proposal in this direction is the access control model developed in the framework of the Author-$\mathcal{X}$ project [5], which provides the support for both access control as well as dissemination policies. Policies are specified in XML and contain information about which subjects can access which portions of the documents. Subjects are qualified by means of credentials, specified using XML. In [5] algorithms for access control as well as computing views of the results are also presented. In addition, architectures for securing XML documents are also discussed. In [3] the authors go further and describe how XML documents may be securely published on the web. The idea is for owners to publish documents, subjects to request access to the documents, and untrusted publishers to give the subjects the views of the documents they are authorized to see, making at the same time the subjects able to verify the authenticity and completeness of the received answer.

The W3C [15] is also specifying standards for XML security. The XML security project is focusing on providing the implementation of security standards for XML. The focus is on XML-Signature Syntax and Processing, XML-Encryption Syntax and Processing, and XML Key Management. While the standards are focusing on what can be implemented in the near-term lot of research is needed on securing XML documents. The work reported in [5] is a good start.

Berners Lee who coined the term semantic web (see [2]) has stressed that the key to developing a semantic web is efficiently managing RDF documents. That is, RDF is fundamental to the semantic web. While XML is limited in providing machine understandable documents, RDF handles this limitation. As a result, RDF provides better support for interoperability as well as searching and cataloging. It also describes contents of documents as well as relationships between various entities in the document. While XML provides syntax and notations, RDF supplements this by providing semantic information in a standardized way. Now to make the semantic web secure, we need to ensure that RDF

documents are secure. This would involve securing XML from a syntactic point of view. However with RDF we also need to ensure that security is preserved at the semantic level. The issues include the security implications of the concepts resource, properties and statements that are part of the RDF specification. That is, how is access control ensured? How can one provide access control at a fine granularity level? What are the security properties of the container model? How can bags, lists and alternatives be protected? Can we specify security policies in RDF? How can we solve semantic inconsistencies for the policies? How can we express security constraints in RDF? What are the security implications of statements about statements? How can we protect RDF schemas? These are difficult questions and we need to start research to provide answers. XML security is just the beginning. Securing RDF is much more challenging.

Another aspect of web data management is managing ontology databases. Now, ontologies may be expressed in RDF and related languages. Therefore, the issues for securing ontologies may be similar to securing RDF documents. That is, access to the ontologies may depend on the roles of the user, and/or on the credentials he or she may possess. On the other hand, one could use ontologies to specify security policies. That is, ontologies may help in securing the semantic web. We need more research in this area.

3.3 Privacy for Web Databases

Privacy is about protecting information about individuals. Privacy has been discussed a great deal in the past especially when it relates to protecting medical information about patients. Social scientists as well as technologists have been working on privacy issues. However, privacy has received enormous attention during the past year. This is mainly because of the advent of the web and now the semantic web, counter-terrorism and national security. For example, in order to extract information from databases about various individuals and perhaps prevent and/or detect potential terrorist attacks, data mining tools are being examined. We have heard a lot about national security vs. privacy in the media. This is mainly due to the fact that people are now realizing that to handle terrorism, the government may need to collect data about individuals and mine the data to extract information. This is causing a major concern with various civil liberties unions. In this section, we discuss privacy threats that arise due to data mining and the semantic web. We also discuss some solutions and provide directions for standards.

Data mining, national security, privacy and web databases. With the web there is now an abundance of data information about individuals that one can obtain within seconds. The data could be structured data or could be multimedia data. Information could be obtained through mining or just from information retrieval. Data mining is an important tool in making the web more intelligent. That is, data mining may be used to mine the data on the web so that the web can evolve into the semantic web. However, this also means that

there may be threats to privacy (see [12]). Therefore, one needs to enforce privacy controls on databases and data mining tools on the semantic web. This is a very difficult problem. In summary, one needs to develop techniques to prevent users from mining and extracting information from data whether they are on the web or on networked servers. Note that data mining is a technology that is critical for say analysts so that they can extract patterns previously unknown. However, we do not want the information to be used in an incorrect manner. For example, based on information about a person, an insurance company could deny insurance or a loan agency could deny loans. In many cases these denials may not be legitimate. Therefore, information providers have to be very careful in what they release. Also, data mining researchers have to ensure that privacy aspects are addressed. While little work has been reported on privacy issues for web databases we are moving in the right direction. As research initiatives are started in this area, we can expect some progress to be made. Note that there are also social and political aspects to consider. That is, technologists, sociologists, policy experts, counter-terrorism experts, and legal experts have to work together to develop appropriate data mining techniques as well as ensure privacy. Privacy policies and standards are also urgently needed. That is, while the technologists develop privacy solutions, we need the policy makers to work with standards organizations (i.e., W3C) so that appropriate privacy standards are developed.

Solutions to the privacy problem for web databases. As we have mentioned, the challenge is to provide solutions to enhance national security as well as extract useful information but at the same time ensure privacy. There is now research at various laboratories on privacy enhanced/sensitive data mining (e.g., Agrawal at IBM Almaden, Gehrke at Cornell University and Clifton at Purdue University, see for example [1], [7], [8]). The idea here is to continue with mining but at the same time ensure privacy as much as possible. For example, Clifton has proposed the use of the multiparty security policy approach for carrying out privacy sensitive data mining. While there is some progress we still have a long way to go. Some useful references are provided in [7]. We give some more details on an approach we are proposing. Note that one mines the data and extracts patterns and trends. The idea is that *privacy constraints* determine which patterns are private and to what extent. For example, suppose one could extract the names and healthcare records. If we have a privacy constraint that states that names and healthcare records are private then this information is not released to the general public. If the information is semi-private, then it is released to those who have a need to know. Essentially, the inference controller approach we have proposed in [14] is one solution to achieve some level of privacy. It could be regarded to be a type of privacy sensitive data mining. In our research we have found many challenges to the inference controller approach. These challenges will have to be addressed when handling privacy constraints (see also [13]). For example, there are data mining tools on the web that mine web databases. The privacy controller should ensure privacy preserving data mining. Ontologies may

be used by the privacy controllers. For example, there may be ontology specification for privacy constructs. Furthermore, XML may be extended to include privacy constraints. RDF may incorporate privacy semantics. We need to carry out more research on the role of ontologies for privacy control. Much of the work on privacy preserving data mining focuses on relational data. We need to carry out research on privacy preserving web data mining which contains unstructured data. We need to combine techniques for privacy preserving data mining with techniques for web data mining to obtain solutions for privacy preserving web data mining.

4 Security and Privacy for Web Services

Security and privacy concerns related to web services are receiving today growing attention from both the industry and research community [9]. Although most of the security and privacy concerns are similar to those of many web-based applications, one distinguishing feature of the Web Service Architecture is that it relies on a repository of information, i.e., the UDDI registry, which can be queried by service requestors and populated by service providers. Even if, at the beginning, UDDI has been mainly conceived as a public registry without specific facilities for security and privacy, today security and privacy issues are becoming more and more crucial, due to the fact that data published in UDDI registries may be highly strategic and sensitive. For instance, a service provider may not want that the information about its web services are accessible to everyone, or a service requestor may want to validate the privacy policy of the discovery agency before interacting with this entity. In the following, we thus mainly focus on security and privacy issues related to UDDI registries management. We start by considering security issues, then we deal with privacy.

4.1 Security for Web Services

When dealing with security, three are the main issues that need to be faced: *authenticity*, *integrity*, and *confidentiality*. In the framework of UDDI, the authenticity property mainly means that the service requestor is assured that the information it receives from the UDDI comes from the source it claims to be from. Ensuring integrity means ensuring that the information are not altered during its transmission from the source to the intended recipients and that data are modified according to the specified access control policies. Finally, confidentiality means that information in the UDDI registry can only be disclosed to requestors authorized according to some specified access control policies. If a two-party architecture is adopted, security properties can be ensured using the strategies adopted in conventional DBMSs [6], since the owner of the information (i.e., the service provider) is also responsible for managing the UDDI. By contrast, such standard mechanisms must be revised when a third-party architecture is adopted. The big issue there is how the provider of the services can

ensure security properties to its data, even if the data are managed by a discovery agency. The most intuitive solution is that of requiring the discovery agency to be trusted with respect to the considered security properties. However, the main drawback of this solution is that large web-based systems cannot be easily verified to be trusted and can be easily penetrated. The challenge is then how such security properties can be ensured without requiring the discovery agency to be trusted.

In the following, we discuss each of the above-mentioned security properties in the context of both a two-party and a third-party architecture.

Integrity and confidentiality. If UDDI registries are managed according to a two-party architecture, integrity and confidentiality can be ensured using the standard mechanisms adopted by conventional DBMSs [6]. In particular, an *access control mechanism* can be used to ensure that UDDI entries are accessed and modified only according to the specified access control policies. Basically, an access control mechanism is a software module that filters data accesses on the basis of a set of access control policies. Only the accesses authorized by the specified policies are granted. Additionally, data can be protected during their transmission from the data server to the requestor using standard encryption techniques [10].

If a third-party architecture is adopted, the access control mechanism must reside at the discovery agency site. However, the drawback of this solution is that the discovery agency must be trusted. An alternative approach to relax this assumption is that of using a technique similar to the one proposed in [5] for the secure broadcasting of XML documents. Basically, the idea is that the service provider encrypts the entries to be published in an UDDI registry according to its access control policies: all the entry portions to which the same policies apply are encrypted with the same key. Then, it publishes the encrypted copy of the entries to the UDDI. Additionally, the service provider is responsible for distributing keys to the service requestors in such a way that each service requestor receives all and only the keys corresponding to the information it is entitled to access. However, exploiting such solution requires the ability of querying encrypted data.

Authenticity. The standard approach for ensuring authenticity is using digital signature techniques [10]. To cope with authenticity requirements, the latest UDDI specifications allow one to optionally sign some of the elements in a registry, according to the W3C XML Signature syntax [15]. This technique can be successfully employed in a two-party architecture. However, it does not fit well in the third-party model, if we do not want to require the discovery agency be trusted wrt authenticity. In such a scenario, it is not possible to directly apply standard digital signature techniques, since a service requestor may require only selected portions of an entry, depending on its needs, or a combination of information residing in different data structures. Additionally, some portions of the requested information could not be delivered to the requestor because of access constraints stated by the specified policies. A solution that can be exploited

in this context (which has been proposed in [4]) is that of applying to UDDI entries the authentication mechanism provided by Merkle hash trees. The approach requires that the service provider sends the discovery agency a summary signature, generated using a technique based on Merkle hash trees, for each entry it is entitled to manage. When a service requestor queries the UDDI registry, the discovery agency sends it, besides the query result, also the signatures of the entries on which the enquiry is performed. In this way, the requestor can locally recompute the same hash value signed by the service provider, and by comparing the two values it can verify whether the discovery agency has altered the content of the query answer and can thus verify its authenticity. However, since a requestor may be returned only selected portions of an entry, it may not be able to recompute the summary signature, which is based on the whole entry. For this reason, the discovery agency sends the requestor a set of additional hash values, referring to the missing portions, that make it able to locally perform the computation of the summary signature. We refer the interested readers to [4] for the details of the approach.

4.2 Privacy for Web Services

To enable privacy protection for web services consumers across multiple domains and services, the World Wide Web Consortium working draft *Web Services Architecture Requirements* has already been defined some specific privacy requirements for web services [15]. In particular, the working draft specifies five privacy requirements for enabling privacy protection for the consumer of a web service across multiple domains and services:

- the WSA must enable privacy policy statements to be expressed about web services;
- advertised web service privacy policies must be expressed in P3P [15];
- the WSA must enable a consumer to access a web service's advertised privacy policy statement;
- the WSA must enable delegation and propagation of privacy policy;
- web services must not be precluded from supporting interactions where one or more parties of the interaction are anonymous.

Most of these requirements have been recently studied and investigated in the W3C P3P Beyond HTTP task force [15]. Further, this task force is working on the identification of the requirements for adopting P3P into a number of protocols and applications other than HTTP, such as XML applications, SOAP, and web services. As a first step to privacy protection, the W3C P3P Beyond HTTP task force recommends that discovery agencies have their own privacy policies that govern the use of data collected both from service providers and service requestors. In this respect, the main requirement stated in [15] is that collected personal information must not be used or disclosed for purposes other than performing the operations for which it was collected, except with the consent of the subject or as required by law. Additionally, such information must be retained only as long as necessary for performing the required operations.

5 Towards a Secure Semantic Web

For the semantic web to be secure all of its components have to be secure. These components include web databases and services, XML and RDF documents, and information integration services. As more progress is made on investigating the various security issues for these components, then we could envisage developing a secure semantic web. Note that logic, proof and trust are at the highest layers of the semantic web. Security cuts across all layers and this is a challenge. That is, we need security for each of the layer and we must also ensure secure interoperability. For example, consider the lowest layer. One needs secure TCP/IP, secure sockets, and secure HTTP. There are now security protocols for these various lower layer protocols. One needs end-to-end security. That is, one cannot just have secure TCP/IP built on untrusted communication layers. That is, we need network security. Next layer is XML. One needs secure XML. That is, access must be controlled to various portions of the document for reading, browsing and modifications. There is research on securing XML. The next step is securing RDF. Now with RDF not only do we need secure XML, we also need security for the interpretations and semantics. For example, under certain contexts, portions of the document may be Unclassified while under certain other context the document may be Classified. As an example, one could declassify an RDF document, once the war is over. Once XML and RDF have been secured the next step is to examine security for ontologies and interoperation. That is, ontologies may have security levels attached to them. The challenge is how does one use these ontologies for secure information integration. Researchers have done some work on the secure interoperability of databases. We need to revisit this research and then determine what else needs to be done so that the information on the web can be managed, integrated and exchanged securely. Closely related to security is privacy. That is, certain portions of the document may be private while certain other portions may be public or semi-private. Privacy has received a lot of attention recently partly due to national security concerns. Privacy for the semantic web may be a critical issue, That is, how does one take advantage of the semantic web and still maintain privacy and sometimes anonymity. We also need to examine the inference problem for the semantic web. Inference is the process of posing queries and deducing new information. It becomes a problem when the deduced information is something the user is unauthorized to know. With the semantic web, and especially with data mining tools, one can make all kinds of inferences. That is the semantic web exacerbates the inference problem. Security should not be an afterthought. We have often heard that one needs to insert security into the system right from the beginning. Similarly, security cannot be an after-thought for the semantic web. However, we cannot also make the system inefficient if we must guarantee one hundred percent security at all times. What is needed is a flexible security policy. During some situations we may need one hundred percent security while during some other situations say thirty percent security (whatever that means) may be sufficient.

6 Conclusions

In this paper we have focused on security and privacy issues for the semantic web. In particular, we have discussed these issues for two of the key components of semantic web, that is, web databases and services. Besides providing background information on web databases and services, we have discussed the main issues related to security and privacy: which are the main challenges, and which are the most promising solutions. Finally, we have discussed some of the issues in developing a secure semantic web.

References

1. Agrawal, R., Srikant, R.: Privacy-preserving Data Mining, Proceedings of the ACM SIGMOD Conference (2000), Dallas, TX, USA.
2. Berners Lee, T., et al.: The Semantic Web (2001), Scientific American.
3. Bertino, E., Carminati, B., Ferrari, E. Thuraisingham, B., Gupta, A.: Selective and Authentic Third-party Distribution of XML Documents. IEEE Transactions on Knowledge and Data Engineering, to appear.
4. Bertino, E., Carminati, B., Ferrari, E.: A Flexible Authentication Method for UDDI Registries, Proceedings of the ICWS Conference, (2003), Las Vegas, Nevada, USA.
5. Bertino, E., Ferrari, E.: Secure and Selective Dissemination of XML Documents. ACM Transactions on Information and System Security, **5(3)** (2002) 290-331.
6. Castano, S., Fugini, M.G., Martella, G., Samarati, P. : Database Security (1995), Addison-Wesley.
7. Clifton, C., Kantarcioglu, M., Vaidya, J.: Defining Privacy for Data Mining, Proceedings of the Next Generation Data Mining Workshop (2002), Baltimore, MD, USA.
8. Gehrke, J.: Research Problems in Data Stream Processing and Privacy-Preserving Data Mining, Proceedings of the Next Generation Data Mining Workshop (2002), Baltimore, MD, USA.
9. IBM Corporation: Security in a Web Services World: A Proposed Architecture and Roadmap, White Paper, Version 1.0, 2002. Available at: `www-106.ibm.com/developerworks/library/ws-secroad/`.
10. Stallings, W.: Network Security Essentials: Applications and Standards (2000), Prentice Hall.
11. Pollmann, C.G.: The XML Security Page. Available at: `http://www.nue.et-inf.uni-siegen.de/~geuer-pollmann/xml_security.html`.
12. Thuraisingham, B.: Web Data Mining: Technologies and Their Applications to Business Intelligence and Counter-terrorism (2003), CRC Press.
13. Thuraisingham, B.: Privacy Constraint Processing in a Privacy Enhanced Database System, Data and Knowledge Engineering, to appear.
14. Thuraisingham B., Ford, W.: Security Constraint Processing in a Distributed Database Management System, IEEE Transactions on Knowledge and Data Engineering (1995).
15. World Wide Web Consortium: www.w3c.org.
16. Universal Description, Discovery and Integration (UDDI): UDDI Version 3.0, UDDI Spec Technical Committee Specification, July, 19th, 2002. Available at: `http://uddi.org/pubs/uddi-v3.00-published-20020719.htm`.