December 2004

Managing Web Services Security

Kenny Khoo
University of Maryland

Lina Zhou
University of Maryland, Baltimore County

Managing Web Services Security

Kenny Khoo
University of Maryland, Baltimore County
Kkhoo1@umbc.edu

Lina Zhou
University of Maryland, Baltimore County
zhoul@umbc.edu

ABSTRACT

The promising features of Web services also make them vulnerable to new types of security threats. Web service providers must assure their clients confidentiality, integrity and availability over a trusted relationship that may be asynchronous and that may involve multiple business partners. Despite the continued significance of the traditional approaches to securing content, transmission and connection in a Web-based business environment, including Secure Socket Layer, Virtual Private Networks, Internet Protocol Security, and so on, they are not able to address the new challenges posed by Web services. This paper aims to provide insight into the management of Web services security. It first introduces the key concepts and reviews state-of-the-art Web services security standards. Then, it aligns the Web services security standards with security threats to provide guidance for the practical implementation of Web services security. Finally, it points out some limitations in the current practice and highlights the managerial implications.

Keywords

XML, SOAP, WS-Security, Web Services, Security

INTRODUCTION

Web services provide a new channel for conducting business (Hanna, 2003). There is a momentum for businesses to adopt web services as part of business to business transactions and to develop new business models (Chen, 2003). According to a survey on senior-level IT executives, engineers and project managers at global 2000 organizations (Netegrity, 2003), 75 percent of the respondents planned to roll out Web services in the next 12 months using a number of security standards. It was also reported in the same survey that more than 40% plan to use four or more standards with a majority emphasizing WS-Security (Khaler, 2002) and Security Assertion Mark-up Language (SAML) (OASIS, 2003). XML Encryption (Reagle, 2002), XML Signature (W3Ca, 2002), Kerberos, and X.509 were among the other standards that the respondents plan to use. However, in the same survey, more than half won't use Web services outside the corporate firewall until they are sure that transactions are protected against cyber-attacks. It raises serious concerns over the security issues associated with adopting Web services.

Web services extend the current client-server model with the concept of "loose coupling," which allow services to be discoverable, platform independent, and expressible with self-describing interface (W3Cb, 2002). It provides a means of communication and interoperability among different software applications that may run on different platforms. However, these applications bring risks to business providers by potentially exposing internal business processes and confidential data to distributed clients. An example of such an application is stock trading (Long, Yuan and Whinston, 2003). The business of stock trading is highly time-sensitive and dependent upon the information-intensive transactions. Therefore, traders expect to have access to trading processes even while traveling (either wirelessly or with fixed-line), international brokers have the need to trade simultaneously in multi-nation markets, and currency traders typically diversify their portfolio by trading over international exchanges. Large volumes of information are being exchanged at any time in the stock market. On the one hand, most of the information such as account number, personal information, and gain/loss from completed transactions should be kept confidential from the third parties. On the other hand, the stock trading business strives to ensure that traders are really who they are in conducting the transactions. All these security issues pose challenges to successfully deploying Web services.

The safety of information exchange can be assured by addressing five security requirements: authentication, authorization, Non-repudiation, confidentiality and Integrity (Nakamur, Hada and Neyama, 2002). Client-to-server and server-to-server authentication must be seamless. If a web service provider receives a request for service, it must be able to verify the requester's identity and privileges. Conversely, the requester who receives any information or service can verify that the information is coming from a trusted source. Based on a requester's authentication, authorization may be granted such that

information meant for a particular client can only be accessed specifically by that client. Because of the multiple hops in a web service environment, it is possible that they do not share the same authentication or authorization infrastructure. SAML, which is an XML based infrastructure, may be implemented to handle this challenge. In non-repudiation, clients should not be able to reject transactions that were initiated by them. Confidentiality ensures that the exchanged information is protected against interception. Integrity provides the assurance that the message was not modified deliberately or accidentally during transit.

Security measures in a Web services environment should be implemented to ensure that data can only be accessed by authorized users, or to provide a certain level of assurance on the identity of service processes when a client is about to pass sensitive information. The rest of paper is organized as follows. We first review Web services standards. Then, we evaluate security challenges specific to Web services and the Web services security standards. Based on our investigation, we provide a guideline for matching security standards to business needs. Finally, we discuss the limitations in the current practice of Web service security and suggest future directions.

WEB SERVICES STANDARDS

The current web environment is "human-centric" in that a typical user would interact with the merchant web-site which lacks integration with the entire supply chain. Web services offer the capability of integrating the processes and becoming more "application-centric". The "application-centric" approach will allow various business applications from different service providers to share data using XML standards. Web services typically consist of four main components: Web service consumer, Web service provider, business agreement, and registry, as shown in Figure 1. The Web service consumer or client program makes a request to, and gets a response from one or several Web service providers. The Web service provider is a company or organization hosting a program in Web service that can receive requests from and send response to Web services consumers. A business agreement or document defines the functionality of a Web service and how that functionality can be accessed by consumers. The fourth component is a registry where Web service providers publish their services and Web consumers search for a web service.

Web services must be published to the Internet community by the Web service provider in order to make them available to potential consumers. Web Services Description Language (WSDL) is a standard XML based vocabulary used to describe Web services, which are then published using the Universal Description, Discovery and Integration (UDDI) protocol, another part of the XML vocabulary. The purpose of WSDL is to provide a file that a client can use to invoke the Web service and its functions. A WSDL document simply consists of a set of definitions. The UDDI protocol is one of the major building blocks required for successful Web services (UDDI, 2001). The UDDI enables businesses to quickly, easily, and dynamically discover and interact with one another independent of the platform, by using applications that they are familiar with. In addition, the UDDI registry allows virtual business functions to be integrated regardless of the internal processes of each business entity as long as they meet the Web services standards. Simple Object Access Protocol (SOAP) is a lightweight, XML based messaging protocol framework for building and exchanging distributed, structured information in a decentralized and distributed environment. There are three parts in a soap message -an envelope, a set of encoding rules, and a convention for representing remote procedure calls and responses.

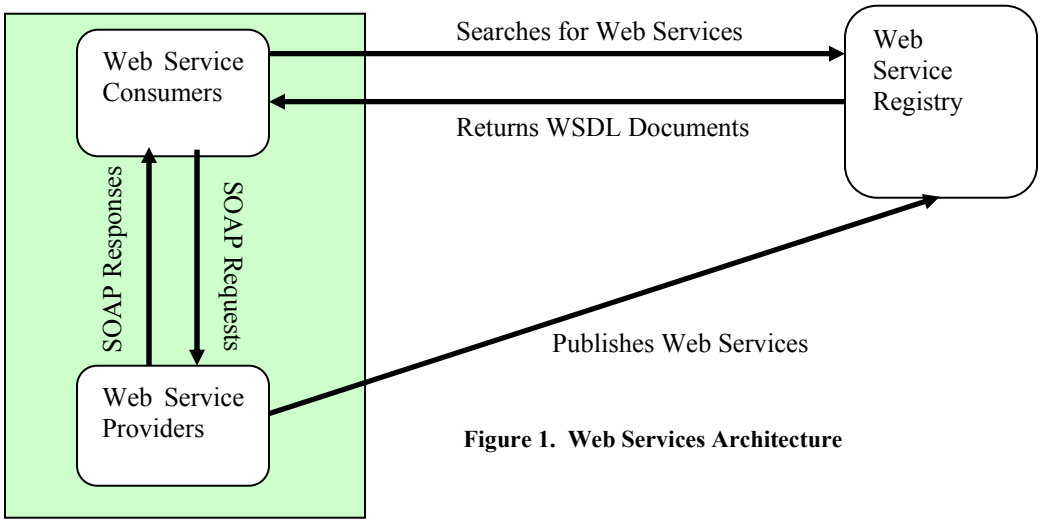

Figure 1. Web Services Architecture

Most of the Web services transactions are messaging based. SOAP has been extensively used as the messaging protocol. In the next section, we will primarily focus on the security issues involved in the SOAP messaging between Web service consumers and Web service providers, as highlighted in Figure 1.

WEB SERVICES SECURITY CHALLENGES

In a Web services environment, multiple trading partners are accessing applications across disparate corporate firewalls using multiple devices and involving multi-step processes (Chang, Chen and Hsu, 2003). Compared with regular business transactions on the Web, Web services pose new challenges to security management, which are selectively listed as follows:

- A security method should support single sign-on schemes. Without such mechanisms in place, each trading partner has to maintain its own authentication and authorization, which may greatly compromise the convenience of Web services.
- A security method needs to consider the security implications of supporting multiple devices (e.g. Personal Digital Assistants, 3G cell phones). For example, wireless standards such as GSM and WAP do not offer end-to-end security (Claessens, Preneel, and Vandewalle, 2001).
- A security method should ensure confidentiality and integrity of the transactions in a multi-step process.

The above security challenges specific to Web services are not addressed by the current point-to-point content security technologies such as SSL, VPNs, and IPSec. Among a number of security implications resulting from Web services, we select information integrity and trust management to illustrate the limitations of traditional security standards.

Information integrity provides the assurance that messages were not modified deliberately or accidentally during transit. A traditional environment may involve only two partners. SSL supports transport layer security between two SSL-enabled parties. It is based on point-to-point connection sessions, and thus each SSL session is unique. When the data are not "in transit" between the two parties, it is not encrypted and thus not secure. The data is vulnerable to attack when it is between two SSL sessions in a multi-party or multi-step operation. Therefore, SSL is not sufficient for ensuring integrity in a multi-party or multi-step Web service transaction. Another drawback of SSL is that it is designed to encrypt the entire document before transmitting it. However, in a Web services environment, there may be a need to only secure part of a document. For example, a merchant should only be allowed to read information in a message that is pertinent to them. Other point-to-point technologies such as VPN and IPSec suffer from the same inherent weaknesses.

Web services across enterprises are dealing with un-trusted clients. Trust management in a multiple-partner transaction must be more dynamic in order for distributed networking to scale (Morioka, Yonemoto, Suzuki and Etoh, 2003). By definition, Web services are loosely coupled. As such, authorization policies are more difficult to implement by individual merchants. Nonetheless, there is still a need to identify if a caller is authorized to request for a Web service. While the commonly used trusted third party approach on the Web (Chang et al, 2003) allows end-to-end agreement with security policies, it does not manage versioning for a long-duration operation. It has been recently proposed that Web services themselves may be used to provide trust services (Baldwin, Shiu and Mont, 2002)

In sum, Web services require a finer grained security protocol beyond the traditional security standards. In the next section, we will discuss Web services security standards that are developed to resolve some of the security challenges.

WEB SERVICES SECURITY STANDARDS

Web services are situated at the application layer within the network protocol stack. Therefore, it is important to secure the perimeter, network, and host to reduce threats. Since Web services are XML based, in principle, existing security measures should be supplemented with XML security frameworks to ensure that Web services are secured. We summarize some of the existing key XML frameworks for Web services security in Table 1. As Web services still rely on the Internet to operate, the non-XML frameworks, which are situated underneath the application layer, can still be used to secure the communication of Web services. Thus, we also identified some non-XML frameworks in Table 1. Next, we elaborate on the key features of each of the XML framework based security standards respectively.

Security Assertion Markup Language (**SAML**) defines an XML-based protocol by which clients can request assertions from SAML authorities and receive responses from them. SAML assertions can be digitally signed. The assertions can convey information about whether accesses to resources are allowed. It allows disparate security systems to interoperate with each other. However, SAML assumes trust between the participants and it defers this responsibility to XML Encryption and XML Digital Signature. There are three kinds of SAML assertion statements: authentication, attribute, and authorization decision. Electronic Business XML (ebXML) is an initiative between OASIS and the United Nations Center for Trade Facilitation and

Electronic Business (UN/CEFACT). ebXML specifies a framework for the exchange of electronic business data. Extensible Access Control Markup Language (**XACML**) is an XML specification for expressing policies for information access over the Internet. It provides a fine-grained access control to XML documents and other documents shared on the internet. XACML differs from SAML in that SAML assertions are formulated at runtime. Extensible Rights Markup Language (**XrML**) provides a method for specifying and managing the rights and policies associated with digital content and services. Compared with XACML, XrML is easier to use but at the cost of expressivity and flexibility.

XML Frameworks	Non-XML Frameworks
High-Level Security Frameworks (SAML, XrML, XACML etc.) WS-Security Simple Object Access Protocol XML Signature XML Encryption XML Firewall XKMS	Transport-Level Security/Secure Socket Layer (TLS/SSL) Transport Layer (HTTP, FTP, SMTP, etc.) Transmission Control Protocol and Internet Protocol (TCP/IP)

Table 1. Web Services Security Specifications

WS-Security (Khaler, 2002) describes the enhancements that can be used to accommodate the current variety of security models and encryption technologies. The goal is achieved by providing quality of protection of SOAP messaging. Web services have typically relied on traditional security methods such as SSL and firewalls. However, SOAP messages can easily pass through firewalls and therefore requires additional security solutions. This can be achieved with WS-security. The WS-Security specification proposes a set of SOAP extensions that can be used when building secure Web services to ensure integrity and confidentiality. The purpose of WS-Security is to allow SOAP messages to be constructed securely. It also supports a variety of security models including Public Key Infrastructure (PKI), Kerberos, and SSL. Among others, security token propagation, message integrity, and message confidentiality are three mechanisms provided by this specification.

Data can be digitally signed. An **XML Signature** is defined within the <Signature> element. XML Signatures provide integrity, message authentication, and/or signer authentication services for data of any type, either located within the XML including the signature or elsewhere. It also provides XML-compliant syntax for representing the signature for Web resources and portions of protocol messages and procedures for computing and verifying such signatures. XML signatures provide proof for non-repudiation of who created it (Naedele, 2003). **XML Encryption** is a process of encrypting data and representing the result with XML. The main element is <Encrypted Data>. XML Encryption defines how digital content is encrypted and decrypted, how the encryption key is passed to a recipient and how encrypted XML data and non-XML data is identified. The result is an XML encryption element that contains or references the cipher data. Both XML Encryption and XML Signature can be applied in any sequence. If XML encryption is applied after XML signature, the XML signature will not be immediately verifiable. Such a document needs to be decrypted before the signature can be verified. The XML Signature specification includes information to identify documents that are encrypted either after signing or before signature. XML Encryption and XML Signature also provide the ability to encrypt only selected portions of a document. This is an enhancement over SSL which encrypts the entire document before it is transmitted.

Current corporate firewalls can only filter at the packet level but not at the content level. They typically work by blocking TCP ports except for port 80 (HTTP traffic), port 25 (email traffic), and port 443 (HTTPS traffic). This raises a serious security issue as many Web services are designed to pass through port 80. **XML firewalls** are designed to examine XML content of the incoming traffic. XML firewalls may be hardware or software based. XML Key Management Specification (**XKMS**) defines protocols for distributing and registering public keys which may be used in conjunction with XML Signature and XML Encryption standards. It also describes how a client may receive key information from a Web service.

A Web Services Security Scenario

In this section, we use a scenario (Figure 2) to illustrate how the Web services security standards can be applied in a typical Web services environment.

Joe Shopper (either a person or an agent), a Web services consumer, is interested in purchasing some products from MyShopping.com. Joe Shopper submits an order of those products on MyShopping.com's purchasing portal using an HTML form. The portal authenticates the user and creates two SOAP messages, a billing SOAP message to MyBilling.com and a shipping SOAP message to MyShipping.com. MyShopping.com can also be considered as a Web services consumer of the services provided by MyShipping.com and MyBilling.com. The SOAP message to the billing Web service includes a WS-Security header enclosing Joe Shopper's name and the password digest. The SOAP envelope includes the payment information encoded with XML Encryption to hide his credit card number. If the billing is completed successfully, MyBilling.com will return a SOAP message to MyShopping.com with the billing status. Once the billing service is completed successfully, the shipping SOAP message is submitted to MyShipping.com. The SOAP message for the shipping Web service includes a WS-Security header enclosing a SAML assertion, which may be signed using XML Signature. The SOAP envelope includes the list of items to be shipped to the consumer. MyShipping.com returns a SOAP message in response to MyShopping.com once the shipping process is completed.

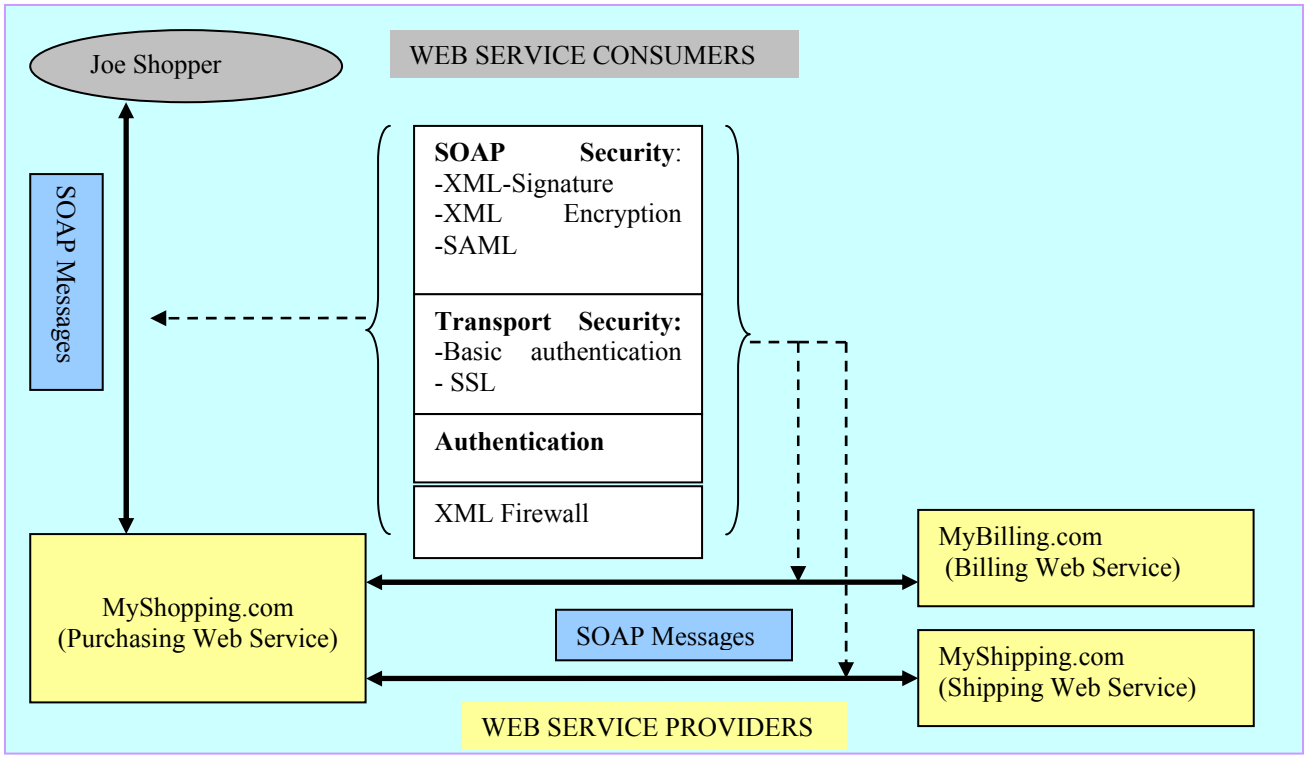

Figure 2. Web Services Scenario Enhanced by Web Services Security

MATCHING SECURITY STANDARDS TO BUSINESS NEEDS

The implications of Web Services innovation for general adopters are significant (Xu , Selsikas and O'Keefe, 2003). Web services can lead to e-service interactions within and across organizations. In view of the unique environment and strategy of individual service providers, security considerations must be customized to meet business goals. There is a wide variety of security threats, including un-authenticated sender, un-authorized receiver, denial of service, dictionary attack, replay attack, token substitution, message modification, man-in-the-middle attack, domain name server attacks, Trojan horse, and so on. Managers should be aware that it is practically impossible to prepare for countering against each and every type of security threat. A cost-effective strategy is to align security-enhancing solutions with business needs. The focus should be placed on reducing the exposure and spread of the risk. To provide some guidance for the selection of XML standards in Web services applications, we match various XML security standards to business security goals in Table 2.

Business managers should be aware of various technologies available to meet the security requirements as they plan for the launch of Web services. Some standards such as XKMS address broader security goals while others such as SAML are more goal-oriented. It is not uncommon for managers to implement more than one of the above standards to ensure Web services security.

XML Standards	Security Goals
WS-Security	Authentication, Confidentiality
SAML	Authentication
XML Digital Signature	Authentication, Integrity, Non-repudiation, Audit, Trust
XML Encryption	Confidentiality, Integrity
XKMS	Confidentiality, Non-repudiation, Audit, Integrity, Trust
XACML	Authorization
XrML	Authorization

Table 2. XML Security Standards and Their Goals

DISCUSSION

Security is a major concern of potential adopters of Web services. Organizations using the traditional HTTPS or TLS/SSL should be aware that their security measures need to be updated to meet the needs of Web services. Moreover, emerging standards for Web services security should be selected cautiously to match them to business security goals. Due to the limitations of the individual Web services security standards, it is necessary to combine multiple standards in implementing Web services security. For example, SAML assumes trust between trading partners, which must be supplemented by XML Signature and XML Encryption. Also, XrML and XAML manage authorization but do not address authentication, which can be complemented with encryption and digital signature protocols. Finally, Web services security must be integrated with the enterprise environments (Nakamur et al, 2002). The recipients of SOAP messages with the accompanying security information should provide an environment where the messages can be processed and acted upon.

We have focused on the implementation of Web service security in this paper. To achieve business success with Web services, an organization should also take into account a number of other factors as follows:

- Financial considerations. Web services will provide a new channel to develop new markets. Gartner predicts that in 2004, sales in the Web services market is expected to grow to $28 billion (Gartner, 2002). However, Web services are geographically independent. It is important that critical business data and proprietary processes are secured from unauthorized access. A breach in corporate data integrity will have serious financial impact.
- Legislative Compliance. Government legislation increasingly requires that consumer data are not revealed without the permission of its owner. Insurance companies, hospitals and organizations working with patient data need to ensure compliance with The Health Insurance Portability and Accountability Act of 1996, known as HIPAA. HIPAA is expected to cost the healthcare industry at least $3.8 billion between 2003 and 2008 (Beaver and Herold, 2003).
- Privacy. Using SOAP messages, data are increasingly being exposed as it moves over the insecure Internet. Any breach of data privacy may result in the loss of trust from consumers and business partners.

CONCLUSION

While traditional security infrastructures can still be used to support Web services, they cannot address the special challenges posed by Web services of allowing multiple parties to access multi-step processes across disparate corporate firewalls using multiple devices. Developing and adopting XML-based web services security standards is therefore important to leverage the capabilities of web services. Faced with a variety of emerging Web services security standards, a service provider should not choose security solutions randomly. The framework for matching the security standards to the business security goals, proposed in this paper, provided some guidance to assist service providers in strategically choosing security solutions.

Web services are already being implemented and are becoming a normal part of daily electronic transactions. Widespread implementation of web services and the generation of new revenue streams rely on ensuring that Web services security requirements are met satisfactorily. Moreover, it is important to emphasize that security of any business application must be integrated into the overall security plan of the firm. Standards for Web service security are still evolving. With the integration of web services security into enterprise security, we can expect to see the roll-out of novel Web services in the future.

REFERENCES

1. Beaver, K. and Herold, R. (2003) Chapter 3, HIPAA Cost Considerations, The Practical Guide to HIPAA Privacy and Security Compliance, Auerbach Publications, Chapter 3.
2. Baldwin, A., Shiu, S. and Mont C, M. (2002) Trust Services: A framework for service-based solutions, *Proceedings of the 26th Annual International Computer Software and Applications Conference* (COMPSAC'02)
3. Chang, S., Chen, Q. and Hsu, M. (2003) Managing Security Policy in a Large Distributed Web Services Environment, *Proceedings of the 27th Annual International Computer Software and Applications Conference* (COMPSAC'03), 617 – 622.
4. Chen, M. (2003) An Analysis of the Driving Forces for the Adoption of Web Services, e-biz Web Workshop, Dec 13-14, Seattle, WA
5. Claessens, J., Preneel, B, and Vandewalle, J. (2001) Combining World Wide Web and Wireless Security, Informatica 26, pp. 123-132.
6. Gartner, (2002) Gartner Says Web Services Will Dominate Deployment of New Application Solutions for Fortune 2000 Companies by 2004, January 14, 2002
7. Hanna, J. (2003) Web Services, Feb3, 2003. Available on the internet at http://hbsworkingknowledge.hbs.edu/pubitem.jhtml?id=3285&sid=-1&t=special_reports_cyber2003
8. Khaler, C. (2002) WS-Security. Available on the internet at http://www-106.ibm.com/developerworks/webservices/library/ws-secure/
9. Long, J., Yuan, M. and Whinston, A., (2003) Securing a New Era of Financial Services, IT Pro, July | August 2003, 15 – 21
10. Morioka, M., Yonemoto, Y., Suzuki, T. and Etoh, M. (2003) Scalable Security Description Framework for Mobile Web Services, *IEEE International Conference on Communications*, 804 – 808.
11. Naedele, M. (2003) Standards for XML and Web Services Security, *Computer*, 36, 4, 96 – 98.
12. Nakamur, Y., Hada, S. and Neyama, R. (2002), Towards the integration of Web Services Security on Enterprise Environments, *Proceedings of the 2002 Symposium on Applications and the Internet* (SAINT'02w), 166 – 175
13. Netegrity, (2003) "Netegrity Web Services Survey Result" Dec 08, 2003. Available on the internet at http://www.netegrity.com/txmindersurvey/TxMSurveyAnalysis.html
14. OASIS, (2003) SAML Version 1.1. Available on the internet at http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security
15. Reagle, J. (2002) XML Encryption Requirements, W3C. Available on the internet at http://www.w3.org/TR/xml-encryption-req
16. UDDI, (2001) UDDI Executive White Paper, Nov. 14, 2001. Available on the internet at http://www.uddi.org/pubs/UDDI_Executive_White_Paper.pdf
17. W3Ca, XML Signature Syntax and Processing, February 2002. Available on the internet at http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/
18. W3Cb Web Services Activity, January 2002. Available on the internet at http://www.w3.org/2002/ws/
19. Xu, H., Seltsikas, P. and O'Keefe, B. (2003), The Implications of Web Services Innovation for General Adopters: Findings and Recommendations, *Proceedings of the Second Workshop on e-Business (Web),* Dec. 13 – 14, 2003, Seattle, WA.