Secure ebXML Business Transaction Models Applying Web Service Security Standards

Dongkyoo Shin and Dongil Shin

Department of Computer Engineering, Sejong University
98 Gunja-Dong, Gwangjin-Gu, Seoul 143-747, Korea
{shindk, dshin}@sejong.ac.kr

Abstract

ebXML is an e-business standard which enables enterprises to exchange business messages, conduct trading relationships, communicate data in common terms and define and register business processes using Web services. ebXML business transaction models are proposed which allow trading partners to securely exchange business transactions by employing Web service security standard technologies. It is shown how each Web service security technology meets the ebXML standard by constructing the experimentation software and validating messages between the trading partners.

Keywords: *electronic commerce, ebXML, Web service security, secure business transaction*

1. Introduction

ebXML (Electronic Business using eXtensible Markup Language) is a suite of standards providing tools for negotiating business processes and contracts using Web services, developed by OASIS (Organization for the Advancement of Structured Information Standards) and UN/CEFACT (United Nations' Center for Trade Facilitation and E-business) [1, 2, 3]. A list of key risks for ebXML is identified as unauthorized transactions and fraud, loss of confidentiality, error detection (application, network/transport, platform), potential loss of management and audit, and potential legal liability [2]. Because ebXML relies on Web service and some of the same underlying HTTP and Web-based architecture as common Web applications, it is susceptible to similar threats and vulnerabilities. Web Service security is based on several important concepts including identification, authentication, authorization, integrity, non-repudiation, confidentiality and privacy [4]. The ebXML suite of standards provides support for security properties in contracts, but it does not fully support automatic security properties negotiation [4, 5].

There are well-known conventional security technologies that can be used by ebXML implementers to resolve the risks [2]: user-id and password, PKI (Public Key Infrastructure), SSL (Secure Socket Layer), S/MIME (Secure Multi-Purpose Internet Mail Extensions). Web Service security technologies emerging recently have extensibility and flexibility suitable for ebXML security implementation such as encryption, digital signature, access control and authentication. There are Web service security standards that deal with security issues related to electronic business as follows [6]. XML digital signatures [7, 8] and SAML (Security Assertion Markup Language) [9, 10] can be exploited to solve the unauthorized transactions and fraud problems in electronic business systems. XML digital signatures are used in ebXML to provide data integrity on messages, existing authentication and authorization schemes as well as

non-repudiation between entities. SAML is recommended to provide identification, authentication and authorization and often used with XACML (eXtensible Access Control Markup Language) [11, 12] to allow or deny access to an XML resource. XML Encryption [13] is recommended to solve the loss of confidentiality problem. Also XKMS (XML Key Management Specification) [14] is recommended for key management as a substitute for PKI.

2. Background

ebXML is a modular suite of specifications for the XML-based global infrastructure for e-business transactions, and provides a standard method to exchange business messages, conduct trading relationships, communicate data in common terms and define and register business processes [1, 2, 3]. The technical infrastructure of ebXML is composed of the following major elements: *Messaging Service*, *Registry*, *Trading Partner Information* (It consists of two specifications: CPP (Collaboration Protocol Profile) and CPA (Collaboration Protocol Agreement)), *Business Process Specification Schema*, *Core Components*.

Some researches on ebXML application have been carried out by many companies, agencies and universities [15]. For example, a research group named CEClD (Center for E-Commerce Infrastructure Development) at the University of Hong Kong developed a platform "Hermes" based on ebXML message specification [16]. The L2L (Library to Library) was proposed and set up as a service integration program based on SOA (Service Oriented Architecture) standards and ebXML, proposed by Sebastian in Ireland University [17]. Recently certified e-Document authority systems are built based on SOA and ebXML standard by some companies and agencies in Korea [18]. The research on the ebXML still goes on all over the world.

3. Secure Business Transaction Models based on the ebXML

A high-level use case scenario for two trading partners is explained based on the ebXML technical architecture specification [1] as follows. *Company A* will first review the contents of an ebXML Registry, especially the registered business processes that may be downloaded or viewed. Based on a review of the information available from an ebXML Registry, *Company A* can build or buy an ebXML implementation suitable for its anticipated ebXML transactions. The next step is for *Company A* to create and register a CPP with the registry. Company A might wish to contribute new business processes to the registry, or simply reference available ones. The CPP will contain the information necessary for a potential partner to determine the business roles in which *Company A* is interested, and the type of protocols it is willing to engage in for these roles. Once *Company A* is registered, *Company B* can look at *Company A*'s CPP to determine that it is compatible with *Company B*'s CPP and requirements. At that point, *Company B* should be able to negotiate a CPA automatically with *Company A*, based on the conformance of the CPPs, plus agreement protocols, given as ebXML standards or recommendations. Finally, the two companies begin actual transactions.

Based on the scenario, we propose two ebXML business transaction models ensuring the trust relationship within the real trading partners. The first model performs a user authentication and updates the CPP in the registries. The second model performs business transactions within the trading partners. These models will explain how each Web service security technology solves the risks for ebXML.

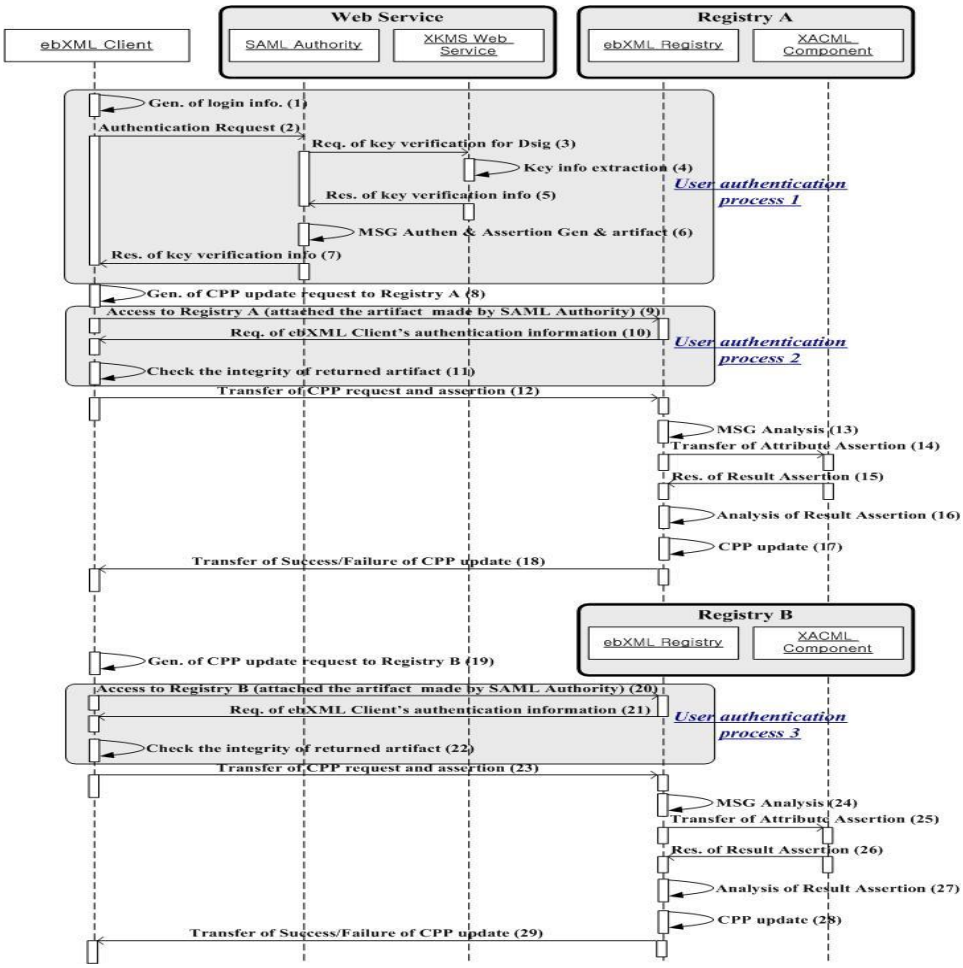


Figure 1. The First Model: Update of CPP (Collaboration Protocol Profile)

In the first model, an ebXML client performs an update for its own CPP in the ebXML registries, where applying security modules to implement business processes satisfies security requirements. The premises for the first model are as follows:

- User registration of *Company A* is completed in the registries, SAML and the XKMS Web Services.
- *Company A* and *B* and the registries have trust relationships with SAML and XKMS Web Services.
- XKMS Web Service has a root role in the CA (Certificate Authority).
- Each CPP of *Company A* and *B* is updated when modification is necessary.
- User based policy documents in XACML format are implemented in each registry.
- Messaging between business entities is based on HTTP-SOAP protocol and XML Signatures and XML Encryptions are applied for secure messaging.

The procedure for the first model is presented in the form of a sequence diagram in Figure 1, where each box in the diagram denotes a Web Service or an application program. Each step denoted by an arrow and number in the diagram is explained as follows:

(1) **Generation of login information:** A Client in *Company A* logs into the local ebXML

intranet system through authentication using user-id and password. An SAML assertion request is generated from this authentication information.

(2) **Authentication request:** Generated SAML assertion is transferred to the SAML Web Service to get an access to registry.

(3) **Request of key verification information for digital signature:** The SAML Web Service requests the client's public key information to XKMS Web Service to verify the received message.

(4) **Extraction of key information:** XKMS Web Service extracts public key information.

(5) **Response of key verification information:** Extracted client's public key information is transferred to the SAML Web Service using response protocol.

(6) **Message authentication and generation of assertion and artifact:** Authentication on the message is performed using the public key information, and then authentication assertion, attribute assertions, and artifact are generated.

(7) **Response of authentication assertion, attribute assertion and artifact:** Generated assertions and artifact are transferred to the client using response protocol.

(8) **Generation of CPP update requests:** Received assertions and CPPs to be updated, and update requests are assembled in the message in the SOAP format.

(9) **Access to Registry A:** An artifact generated by SAML Authority is transferred to Registry A.

(10) **Req. of ebXML Client's authentication information:** To request ebXML Client's authentication information, ebXML Registry of Registry A sends the artifact, which is received from ebXML Client, to ebXML Client.

(11) **Check the integrity of returned artifact:** ebXML Client verifies the integrity of returned artifact from ebXML Registry of Registry A..

(12) **Transfer of CPP updated requests and assertions:** A generated message is transferred to the registry A.

(13) **Message analysis:** The registry A analyzes the received message and perceives the requests. The update of CPP is possible when the user of the client has a role of "ContentOwner". To check the role, the positive response from the XACML Web Service is required.

(14) **Transfer of attribute assertion:** Attribute assertion of the client is transferred to the XACML Web Service.

(15) **Response of result assertion:** Authorization decision assertions are generated and transferred to the registry A, if the attribute assertion meets the XACML policy for documents.

(16) **Analysis of result assertion:** The registry analyzes the response from the XACML Web Service, and proceeds to the CPP update in case it receives authorization decision assertion. Otherwise, it cannot update CPP.

(17) **CPP update:** CPP is updated following the updated request.

(18) **Transfer of success/failure of CPP update:** Message on success/failure of CPP update is transferred to the client.

From (19) to (29) is the same to from (9) to (18).

In the second model, two ebXML client exchange business transactions, where security requirements are satisfied by applying security modules to implement business processes. Additional premises for the second model are as follows: *Company A* and *B* have already exchanged CPA documents and agreed to use XML security technologies

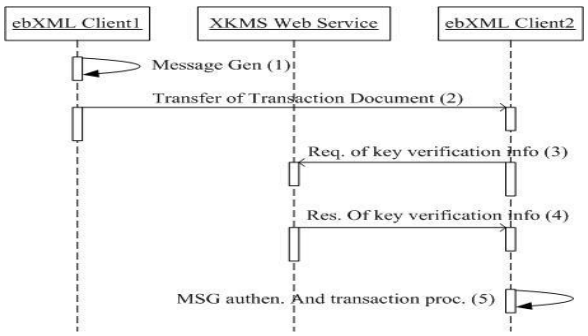


Figure 2. The second model: Exchange of Business Transactions

The procedures for the second model are presented in the form of a sequence diagram in Figure 2. where Client 1 belong to *Company A*, and Client 2 belongs to *Company B*. An arrow denotes each step and number in the diagram and is explained as follows:

(1) **Message generation after CPA analysis:** Each client completes the generation of CPA for a business transaction, and Client 1 creates a transaction document.
(2) **Transfer of transaction document:** The transaction document is transferred from Client 1 to Client 2.
(3) **Request of key verification information for digital signature**: Client 2 requests Client 1's public key information to XKMS Web Service to verify the received message.
(4) **Response of key verification information**: The extracted client's public key information is transferred to Client 2 using response protocol.
(5) **Message Authentication and transaction processing:** Authentication on the transaction message is performed using Client 1's public key information and the transaction is processed.

We verified how each Web service security technology meets the ebXML standard by constructing the experimentation software and validating messages between the trading partners.

4. Conclusion

In this paper, we proposed two business transaction models based on ebXML that allow trading partners to securely exchange business transactions by employing Web service security technologies. We have shown how each XML security technology meets the ebXML standard and solves the risks by checking the messages.

Acknowledgement

This research is supported by Seoul R&BD Program (SS110008).

References

[1] UN/CEFACT and OASIS Technical Specifications, ebXML Technical Architecture Specification, UN/CEFACT and OASIS **(2001)**.
[2] UN/CEFACT and OASIS Technical Reports, ebXML Technical Architecture Risk Assessment V1.0, UN/CEFACT and OASIS, ebXML Security Team, **(2001)**.
[3] S. Patil, E. Newcomer, "ebXML and Web Services", IEEE Internet Comp., Vol. 7, No. 3, **(2003)** pp. 74-82.
[4] A. Singhal, T. Winograd and K. Scarfone, "Guide to Secure Web Services", NIST (National Institute of Standards and Technology) Special Publication 800-95, **(2007)**.
[5] W3C Working Group Note, Web Services Glossary, W3C, February 11 **(2004)** http://www.w3.org/TR/ws-gloss/.

[6] N. A. Nordbotten, "XML and Web Services Security Standards", IEEE Communications Surveys & Tutorials, Vol. 11, Issue 3, **(2009)** pp. 4-21.

[7] W3C Recommendation, XML Signature Syntax and Processing (Second Edition), W3C, **(2008)** http://www.w3.org/TR/xmldsig-core/.

[8] Y. Chen, W. Guo and X. Zhao, "Study of XML digital signature for resource document fragment", 2nd International Conference on Information Science and Engineering (ICISE), Dec**. (2010)** pp. 1541-1544

[9] OASIS Standard, Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS, **(2005)** http://docs.oasis-open.org/security/saml/v2.0/.

[10] P. Harding, L. Johansson and N. Klingenstein, "Dynamic Security Assertion Markup Language: Simplifying Single Sign-On", IEEE Security & Privacy, Vol. 6, Issue 2, **(2008)** pp. 83-85.

[11] OASIS Standard, eXtensible Access Control Markup Language (XACML) Version 2.0 OASIS, **(2005)** http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf.

[12] S. Marouf, M. Shehab, A. Squicciarini, "Adaptive Reordering and Clustering-Based Framework for Efficient XACML Policy Evaluation", IEEE Transactions on Services Computing, Vol. 4, Issue. 4**, (2011)** pp. 300-313.

[13] W3C Recommendation, XML Encryption Syntax and Processing, W3C, **(2002)** http://www.w3.org/TR/xmlenc-core/.

[14] W3C Recommendation, XML Key Management Specification (XKMS 2.0) Version 2.0, W3C, **(2005)** http://www.w3.org/TR/xkms2/.

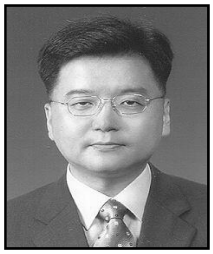
[15] L. Qin and B. Li, "An SOA Architecture with ebXML", 2010 International Conferences on Service Science (ICSS) **(2010)** May 13-14; Hangzhou, China, pp. 358-362.

[16] CECID press release, CECID Hermes H2O Facilitates Reliable Data Exchange for Supply Chain Integration in European Telecommunications Industry, **(2009)** July 10.

[17] Chunwang Li, "A Study on the Standards of SOA", Journal of Modern Library and Information Technology, Vol.5, **(2007)** pp. 1-6.

[18] Certified e-Document Authority, http://www.ceda.or.kr/Eng/.

Authors



Dongkyoo Shin received a B.S. in Computer Science & Statistics from Seoul National University, Korea, in 1986, an M.S. in Computer Science from Illinois Institute of Technology, Chicago, Illinois, in 1992, and a Ph.D. in Computer Science from Texas A&M University, College Station, Texas, in 1997. He is currently a Professor in the Department of Computer Science & Engineering at Sejong University in Korea. From 1986 to 1991, he worked in Korea Institute of Defense Analyses, where he developed database application software. From 1997 to 1998, he worked in the Multimedia Research Institute of Hyundai Electronics Co., Korea as a Principal Researcher. His research interests include XML Security, XML based middleware, multimedia application, biological database, mobile Internet and ubiquitous computing.

Dongil Shin received a B.S. in Computer Science from Yonsei University, Seoul, Korea, in 1988. He received an M.S. in Computer Science from Washington State University, Pullman, Washington, U.S.A., in 1993, and a Ph.D. from University of North Texas, Denton Texas, U.S.A., in 1997. He was a senior researcher at System Engineering Research Institute, Deajun, Korea, in 1997. Since 1998, he has been with the Department of Computer Science & Engineering at Sejong University in Korea where he is currently a Professor. His research interests include Mobile Internet, Computer Supported Cooperative Work, Object-Oriented Database, Distributed Database, Data Mining and Machine Learning.