Developing a Web Service Security Framework

Yangil Park
University of Wisconsin – LaCrosse, USA

Jeng-Chung Chen
National Cheng-Kung University, Taiwan

INTRODUCTION

Web Service (WS) is an open standard software component that uses *Extensible Markup Language* (XML) functions to access and exchange data via networks in communicating with other WSs. In business transactions, Web Service Description Language (WSDL) is used to describe data and deliver all parameters, return values, and types. However, the convenience of using *WSDL* in business transactions also lets hackers snoop and analyze data easily. In addition, the lack of Web Service standards at present makes the *security* issue even more serious in business transactions using WS. This article proposes a high-level *security* model for the *security* problems of the applications. Unlike other studies in the field, this study is dedicated to provide a total solution consisting of technological, organizational, and managerial aspects when using WS. Therefore, the understanding and development of business behaviors is essential in this study. It first introduces current uses of WS. Then definitions of WS, WS security, and WS policy are reviewed. Finally, a WS security model is proposed and explained in the following examples.

BACKGROUND

The Internet is becoming a global common platform where organizations and individuals communicate with each other to carry out various commercial activities and to provide value-added services (Wang, Huang, Qu, & Xie,, 2004). According to the World Wide Web Consortium (W3C, 2004), Web service means that an application program can be described and invoked, and made use of Uniform Resource Identifier (URI) to distinguish via XML. The *application program interfaces* define ways of contacting and supporting other application programs in order to urge directly through the protocol conforming to the Internet with the information of *XML* form. Web service technology allows users to customize services according to their own needs. This allows businesses to interact more accurately and efficiently with customers, cooperative enterprises, and suppliers.

According to a recent Gartner survey, 10 percent of midsize businesses cited using Web services for some production applications, while 47 percent of midsize stated that they plan to deploy Web services (Browning & Anderson, 2004). Also, Gartner Dataquest predicted that Web services will grow from $56 billion worldwide in 2003 to $283 billion worldwide in 2007 (Varbusiness, 2005). By then, Web services will take hold as a competitive differentiator in business relationships and product innovation (Andrews, 2003; Fensel and Bussler, 2002). Enterprises that want to remain competitive will need to use Web services to provide commonly requested data to their partners (Andrews, 2003) and, therefore, Web service technology will no longer offer a competitive advantage to enterprises. It is necessary for them to become competitive (Wiseth, 2004).

When an enterprise has some basic Web services, the high-level functional demands such as the service *security*, the service composition, and the service semantics will increase, and they are critical to the success of deploying Web services (Wang, et al., 2004). Presently, service-oriented architectures use the Web services to work on the business transaction based on the *Web Service Description Language* (WSDL). During the process of digital data delivery hackers are capable of obtaining the parameters. This data can be decoded and analyzed, which could cause a threat to a business. Communication over Web services is done by using a *Simple Object Access Protocol* (SOAP) that is associated with other programs that are built on *XML*. *SOAP* transfers everything over the HTTP, allowing data to pass through firewalls via a TCP port. This enables information to travel through firewalled ports, but this kind of firewall penetration also adds another *security* concern.

Industry observers have said the biggest obstacle to the wider adoption of Web services has been security concerns (Geer, 2003). Thus, security is critical to the adoption of Web services by enterprises, but, as it stands today, the Web service framework does not meet basic *security* requirements (Wang, et al., 2004). With this in mind, this study attempts to develop and implement a web services *security* model and discuss the essential security problems of this structure.

WEB SERVICE SECURITY FRAMEWORK

Web Service

Web services are a relatively new and emerging technology, they allow enterprises to share application logic and data using the standardized data and messaging formats, namely *XML* and *SOAP*. Web services can be accessed through the Internet and are based on existing communication protocols, such as HTTP. However, a standard definition of Web services has yet to be resolved. According to W3C (2004), "a Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically *WSDL*)." In addition, IBM (n.d.) defines "Web service is a new breed of Web application that is self-contained and self-describing, and which can provide functionality and interoperation ranging from the very basic to the most complicated business and scientific processes." Gartner Group (Natis, 2003) defines Web services as a software module that represents a business function (or a business service) and can be accessed by another application (a client, a server, or another Web service) over public networks using generally available ubiquitous protocols and transports (i.e. SOAP over HTTP).

From the above-mentioned definitions, one can understand that the purpose of the Web services is providing an interoperable interface for application-to-application interaction over public networks. The Internet protocols and standards on which Web Services are based are:

- *Simple Object Access Protocol* (SOAP) that enables communications among Web services.
- Web Services Description Language (WSDL) is the *XML* language that providers use to describe their Web Services.
- Universal Description, Discovery and Integration (UDDI) directories enable brokers to register, categorize, and list Web services and requesters to find them.

In traditional application development, programmers spend a great amount of time to tell an application to find another application, and this connection may require maintenance over the course of its lifetime; again, using human application developers. Web services provide certain secure protection in terms of quality, security, data integrity, and complicated transaction. Table 1 presents the comparison of traditional applications and Web services.

Web Service Security

Security is considered the biggest obstacle to general adoption of Web services (Geer, 2003). Cross-enterprise exchange of information over the Internet is vital but may have *security* implications. *Security* issues over the Internet are important, because the Internet is an insecure and non-trustable public network infrastructure, prone to malicious attacks by professional and amateur intruders (Wang, et al., 2004). According to O'Neill (2002) and O'Neill, White, & Watters, (2003), Web services security challenges are as follows:

- The challenge of *security* based on the end user of a Web Service: In order for users to gain access to a

Table 1. Comparison of traditional applications and Web services

Traditional application	Web services
Centralized	Decentralized
Contained and controlled	Open and unmonitored
Limited, defined user base	Unknown, unlimited user base
Secure (risk minimized)	Exposed (open to random events)
Proprietary	Shared
Fixed, well-defined, compiled	Built dynamically, on-the-fly
Incremental scale based on known demand	Unlimited scale, based on unknown, unpredictable demand
Staged, periodic changes	Continuous, *ad hoc* changes
Implementation technologies are all the same or compatible	Heterogeneous in implementation technologies

(Modify from Ratnasingam, 2002; Yang, 2002)