Chapter 8
Engineering Secure Web Services

Douglas Rodrigues
Universidade de São Paulo, Brazil

Kalinka Regina Lucas Jaquie Castelo Branco
Universidade de São Paulo, Brazil

Julio Cezar Estrella
Universidade de São Paulo, Brazil

Nuno Antunes
Universidade de Coimbra, Portugal

Francisco José Monaco
Universidade de São Paulo, Brazil

Marco Vieira
Universidade de Coimbra, Portugal

ABSTRACT

Web services are key components in the implementation of Service Oriented Architectures (SOA), which must satisfy proper security requirements in order to be able to support critical business processes. Research works show that a large number of web services are deployed with significant security flaws, ranging from code vulnerabilities to the incorrect use of security standards and protocols. This chapter discusses state of the art techniques and tools for the deployment of secure web services, including standards and protocols for the deployment of secure services, and security assessment approaches. The chapter also discusses how relevant security aspects can be correlated into practical engineering approaches.

INTRODUCTION

The increasing use of Service-Orient Architectures (SOA) in critical applications demands for dependable and cost-effective techniques to ensure high security. Web services (WS), the cornerstone of the current SOA technology, are widely used for linking suppliers and clients in different sectors such as banking and financial services, transportation, manufacturing, to name a few. However, the problem of engineering secure web services is a non-trivial task as several studies show that a large number of WS implementations are deployed with security flows that range from code vulnerability to inadequate use of standards and protocols.

Engineering secure web services requires developers to clearly identify and understand security requirements. To implement those requirements adequate security standards and protocols have to be applied. While essential WS standards such as XML (eXtensible Markup Language), SOAP

DOI: 10.4018/978-1-4666-4707-7.ch008

(Simple Object Access Protocol), UDDI (Universal Description, Discovery and Integration), WSDL (Web Service Definition Language) approach the basic concepts of interoperable services, the design of secure WS requires complementary rules to be added. Two very well-known examples are the OASIS (Organization for the Advancement of Structured Information Standards) standards WS-Security and SAML (Security Access Markup Language). The former aims at SOAP message security and provides integrity and confidentiality features. The latter focuses on exchanging security information. Developers must therefore understand the main security specifications for Web Services, which include cryptographic algorithms and techniques that implement digital signatures (e.g., WS-Security, WS-Conversation, XML-Signature, XML-Encryption, XACML (OASIS eXtensible Access Control Markup Language), SAML (Security Assertion Markup Language)).

Applying adequate security standards and protocols is not sufficient for guaranteeing secure web services. In fact, software design and coding defects are a major source of vulnerabilities and can put at stake any security countermeasures. For example, interface and communication faults related to problems in the interaction among software components/modules are particularly relevant in service-oriented environments, as services must provide a secure interface to the client applications, even in the presence of malicious inputs. WS developers need to be aware of techniques and tools that help them to assess how secure a service is, such as black-box testing (e.g., robustness testing and penetration testing), and white-box analysis (e.g., code inspection and static code analysis).

This chapter reviews the existing standards, protocols, and tools for developing secure web services, presenting also the most frequent attacks performed against web services and the countermeasures that could be used to avoid them. Additionally, the chapter presents several techniques and tools for assessing the security of web services, which allow checking the effectiveness of the underlying security mechanisms and coding practices.

SECURITY STANDARDS AND PROTOCOLS FOR WEB SERVICES

Enabling information security in the Internet is a mandatory step for fostering business on the Web, especially if we consider systems based on Web services and SOA architectures. In its native form, Web services do not take into account security requirements, which, in most cases, are superficially met by developing security standards in the context of XML-based SOAP messages.

Multi-hop message routing between multiple Web services is commonly used to achieve scalability and also to bridge different protocols. Some technologies such as TLS/SSL – Transport Layer Security/Secure Sockets Layer were initially developed to guarantee the confidentiality between two parties (Dierks and Allen, 1999), (Freier et al., 1996), but they do not provide end-to-end security. To address this challenge, diverse security principles must be applied to different contexts, taking into account both point-to-point and end-to-end settings, as well as the associated considerations concerning the privacy of user information shared in this process. To enable security in this new environment, novel mechanisms have to be put on top of the ones already available at the transport and network layers of the TCP/IP stack. Standards such as XML, SOAP, UDDI and WSDL address the basics of interoperable services, but for secure Web services and SOA other rules must be added and approved (currently a *de facto* security standard for SOA architectures is not available).

Security at the Network Layer

The standard method for providing privacy, integrity and authenticity of information transferred across IP networks is the IPSec protocol (Kent &