Chapter 3.14
Merkle Tree Authentication in UDDI Registries

Elisa Bertino
Purdue University, USA

Barbara Carminati
Università dell'Insubria, Italy

Elena Ferrari
Università dell'Insubria, Italy

ABSTRACT

UDDI registries are today the standard way of publishing information on Web services. They can be thought of as a structured repository of information that can be queried by clients to find the Web services that better fit their needs. Even if, at the beginning, UDDI has been mainly conceived as a public registry without specific facilities for security, today security issues are becoming more and more crucial, due to the fact that data published in UDDI registries may be highly strategic and sensitive. In this paper, we focus on authenticity issues by proposing a method based on Merkle Hash Trees, which does not require the party managing the UDDI to be trusted with authenticity. In the paper, besides giving all the details of the proposed solution, we show its benefit with standard digital signature techniques.

INTRODUCTION

XML Web services are today becoming the platform for application integration and management on the Internet. Basically, an XML Web service is a software service with three main characteristics: *1)* the use of a standard Web protocol (in most cases SOAP [Soap]) to expose the service functionalities; *2)* an XML-based description (through WSDL [Wsdl]) of the interface; and *3)* the use of UDDI [UDDIv3] to publish information regarding the Web service and to make this information available to potential clients. UDDI is an XML-based registry with the primary goal

of making widely available information on Web services. It thus provides a structured and standard description of the Web service functionalities, as well as searching facilities to help in finding the provider(s) that better fit the client requirements. Even if, at the beginning, UDDI has been mainly conceived as a public registry without specific facilities for security, today security issues are becoming more and more crucial, due to the fact that data published in UDDI registries may be highly strategic and sensitive. In this respect, a key issue regards authenticity: it should be possible for a client querying a UDDI registry to first verify that the received answer actually originated at the claimed source, and, then, that the party managing the UDDI registry did not maliciously modify some of its portions before returning them to a client. To deal with this issue, UDDI specifications allow one to optionally sign some of the elements in a registry, according to the W3C XML Signature syntax.

Authenticity issues are particular crucial when UDDI registries are managed according to a third-party architecture. The basic principle of a third-party architecture is the distinction between the owner, who produces the information, and one or more *publishers,* who are responsible for managing (a portion of) the owner information and for answering client queries. Such architectures are today becoming more and more popular, because of their scalability and the ability of efficiently managing a large number of clients and a great amount of data. UDDI can be implemented according to either a third-party or a two-party architecture. A third-party architecture consists of a *service provider,* that is, the owner of the services, the *service requestors,* that is, the parties who request the services, and a *discovery agency,* that is, the UDDI registry. In a two-party architecture, there is no distinction between the service provider and the discovery agency. In this paper we focus on authenticity issues for third-party implementations of UDDI. The main problem is how the owner of the services can ensure the authenticity of its data, even if the data are managed by a third-party (i.e., the discovery agency). The most intuitive solution is that of requiring the discovery agency to be trusted with respect to authenticity. 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. For this reason, in this paper, we propose an alternative approach, which we have previously developed for generic XML data (BCFTG) distributed according to a third-party architecture. The main benefit of the proposed solution is that it does not require the discovery agency to be trusted with authenticity. It is important to remark that in the scenario we consider, it is not possible to directly apply standard digital signature techniques to ensure authenticity, since a client may require only selected portions of a document, depending on its needs, and thus it is not enough that the owner of the data signs each document it sends to the publisher. For this reason, we apply an alternative solution, which requires that the owner sends the publisher, in addition to the information it is entitled to manage, a summary signature, generated using a technique based on Merkle hash trees (Merkle, 1989). The idea is that when a client submits a query to a publisher requiring any portion of the managed data, the publisher sends him/her, besides the query result, also the signatures of the documents on which the query is performed. In this way, the client can locally recompute the same bottom-up hash value signed by the owner, and by comparing the two values he/she can verify whether the publisher has altered the content of the query answer and can thus verify its authenticity. The problem with this approach is that since the client may be returned only selected portions of a document, he/she may not be able to recompute the summary signature, which is based on the whole document. For this reason, the publisher sends the client a set of additional hash