Trust Management for Web Services

Weiliang Zhao and Vijay Varadharajan
Department of Computing
Macquarie University
NSW 2109, Australia
Email: wzhao,vijay@ics.mq.edu.au

Abstract

In this paper, we propose a comprehensive trust management approach for web services that covers the analysis/ modelling of trust relationships and the development of trust management layer in a consistent manner. The specific characteristics of trust relationships in web services are discussed. We introduce a separated trust management layer for web services that can hold computing components for trust management tasks. A trust management architecture for web services is proposed for building up the trust management layer. The proposed trust management architecture for web services deals with trust requirements, trust evaluation, and trust consumption in web services under a unified umbrella and it provides a solid foundation upon which may evolve the trust management layer for web services.

1. Introduction

Web services technologies make distributed computing components to be easily integrated across business boundaries and computing platforms. On the other hand, the web services technologies introduce a high degree of complexity of runtime operations. In web services, different kinds of business partners could be involved, and it may be possible for web services to require other services offered by third parties; the providers of web services may belong to different security domains; and the users of web services may not be predetermined. Undoubtedly, security is a key area to be addressed for delivering integrated, interoperable solutions under web services architecture. There have been many technologies that focus on building blocks and specific aspects of a broad range of security issues [1]. However it is still lacking an integrated solution and architecture that can address access control and related trust management for web services in a consistent manner. In our previous work, we have proposed a comprehensive trust management solution that covers both the analysis/modelling of trust relationships and the development of trust management systems [3, 4, 5]. Employing our previous results as a foundation, we propose a trust management approach for web services in this paper.

The remainder of this paper is organized as follows. Section 2 overviews the taxonomy framework of trust. Section 3 discusses trust relationships in web services. Section 4 proposes a trust management architecture of web services. Section 5 describes web services architecture layers. Section 6 provides concluding remarks for this paper.

2. Taxonomy Framework of Trust

The taxonomy framework of trust provides terminologies and enable tools for the anlysis/modelling of trust relationships in distributed information systems. The taxonomy framework of trust is based on the formal definition of trust relationships and it includes the classification of trust; the properties of trust including trust direction, trust symmetry, scope and diversity of trust relationships; and operations and definitions about the relations of trust relationships. Here we only provide a high level overview about the taxonomy framework of trust. More details about the elements of the taxonomy framework of trust can be found in [3, 5].

3. Trust Relationships in Web Services

We consider some generic requirements for trust relationships about web services. Particularly, in web services paradigm, there are some specific trust relationships that are related to access control of web services or web service methods. The web services or web service methods are the resources to be protected. These trust relationships have a set of specific characteristics that are only existent in the web services paradigm. They capture the requirements for access control of the involved web services or web services methods. Based on the formal model of trust relationship

978-0-7695-3310-0/08 $25.00 © 2008 Crown Copyright
DOI 10.1109/ICWS.2008.53

proposed in our previous work [3], there are the following concerns in the analysis and modelling of trust relationships in the access control for web services:

- The trustor set is normally modelled as a set of web services or a set of web service methods. A specific case is that the trustor set only includes one web service or web service method.
- The trustee set includes the entities that request the web services or web service methods in the trustor set. The trustee set may be composed of a set of web services or other types of requesters.
- The condition set may require the trust mechanisms such as credentials, reputation, data storage, and environment parameters. For web services, security tokens in SOAP messages defined in WS-Trust provide evidences for some conditions in the condition set and WS-Trust is the standard trust mechanism at the messaging level. However, it is possible to have more conditions that are irrelevant to these security tokens.
- The property set may include the basic operations [2] on web services or web services methods such as (1) execute: access and execute the web services or web service methods; (2) update: update web services or web services methods. The executable codes and metadata could be updated. This property is normally useful for administrators and developers for updating computing components; (3) find: provide search capabilities for properties and metadata associated with the web services or web service methods.

Here we only provide a general discussion about trust relationships in web services paradigm. There may be complex situations of trust relationships in web services.

4. Trust Management Architecture for Web Services

In this section, we propose an architecture for building a trust management layer for web services. The architecture will provide a standard, high level design and can be used as an auxiliary tool in the whole life cycle of the development of trust management layer for web services, including specification of requirements, preliminary design, active deployment, and maintenance. This architecture can be used as the basis for dependency and consistency analysis for trust management tasks related with web services. The trust management layer for web services will provide facilities for all the trust management tasks related with trust relationships, in particular, the locating, evaluating, and consuming of trust relationships.

The proposed architecture holds all components of trust management tasks for web services. The architecture is expressed in Figure 1.

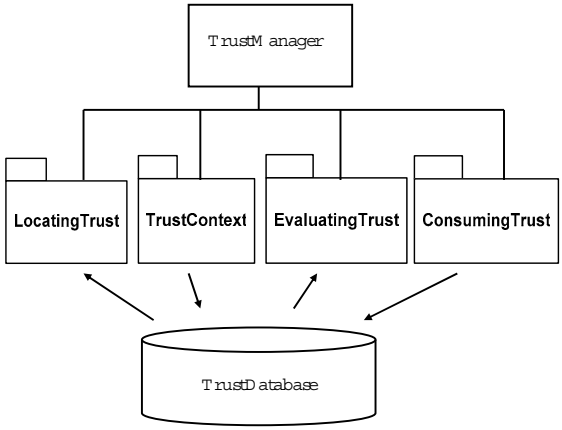

Figure 1. Trust Management Architecture for Web Services

TrustDatabase provides a persistent storage mechanism for storing and retrieving trust related information including trust relationships and trust parameters for web services.

TrustManager is devised as the manager of the trust management layer for web services. It targets the top level management and control of trust management tasks for web services at run time. TrustManger links the functional packages of trust management architecture for web services including LocatingTrust, TrustContext, EvaluatingTrust, and ConsumingTrust.

LocatingTrust is the package for finding the trust relationship and associated processing information based on the request. There are three components in this package (see Figure 2) that are referred to as "Locating Trust Controller", "Trust Relationship Locator", and "Authentication Controller". "Locating Trust Controller" is the management component that receives the trust request and it assigns tasks to "Trust Relationship Locator" and "Authentication Controller". "Trust Relationship Locator" is the component that finds the requested trust relationship and associated processing information from the TrustDatabase. "Authentication Controller" is the component that deals with authentication of the requester and normally it employs existing authentication services in the system to perform the authentication task. The components in this package have been included in the general trust management architecture for distributed information systems [4] and they should be the implementation of corresponding generic components

in the general trust management architecture.

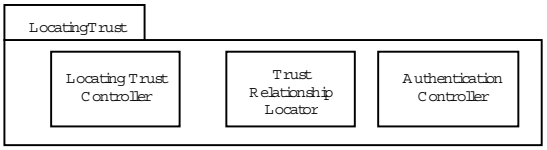

Figure 2. Components of LocatingTrust

TrustContext contains the computing components for the management and processing of contextual information related with trust management of web services.

The handling of WS-Trust SOAP messages that include security tokens and other trust-relevant information is a specific task in web services paradigm. It is necessary to have a computing component in the trust management architecture to adapt the incoming WS-Trust SOAP messages and generate the outgoing WS-Trust SOAP messages. This computing component is devised as "WS-Trust Message Handler".

In web services paradigm, there is a rich messaging environment supported by multiple modular SOAP-based security specifications such as WS-security, WS-Policy, WS-Privacy, WS-SecureConversation, and WS-Federation. These specifications provide building blocks for a broad range of security needs at the messaging level. A broad variety of contextual information of web services is embedded in these building blocks of SOAP-based messages. For web service, the contextual information could be critical to trust decisions. In particular, the contextual information in the building blocks defined by web services security specifications may play an important role in some trust relationships. It is necessary to have a computing component to capture and manage trust-relevant contextual information in web services. This computing component is devised as "Context Information Handler".

The package TrustContext has computing components (see Figure 3) as "TrustContext Controller", "WS-Trust Message Handler" and "Context Information Handler". "TrustContext Controller"is the manager of this package that assigns tasks to "WS-Trust Message Handler" and "Context Information Handler". "WS-Trust Message Handler" adapts the incoming WS-Trust SOAP messages and generates the outgoing WS-Trust SOAP messages. "Context Information Handler" looks after the management of trust-relevant contextual information for trust evaluation and trust consuming in a defined processing scope such as a transaction or a session. These components are specific in the trust management architecture for web services and they have no corresponding components in the general trust management architecture for distributed information systems.

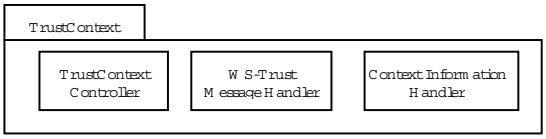

Figure 3. Components of TrustContext

EvaluatingTrust contains computing components required for the evaluation of trust relationships. The evaluation of a trust relationship involves checking whether the conditions of a trust relationship can be satisfied or not. Multiple trust mechanisms can be involved in the evaluation of a single trust relationship. The EvaluatingTrust provides an integration place for multiple trust mechanisms to cooperate with each other.

In EvaluatingTrust (see Figure 4), "Trust Evaluation Controller" is the computing component that assigns evaluation tasks to other functional components in this package. EvaluatingTrust has functional components for specific evaluating tasks, namely "Credential Evaluation", "Reputation Evaluation", "Stored Data Evaluation", and "Environment Evaluation". These components should be the implementation of corresponding generic components of general trust management architecture for distributed information systems [4]. For "Credential Evaluation", some credentials may come from the "WS-Trust Message Handler" in package TrustContext. For "Environment Evaluation", some environment parameters may come from the "Context Information Handler" in package TrustContext.

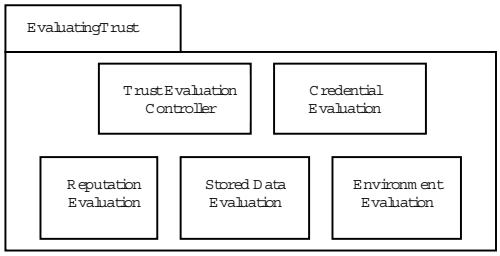

Figure 4. Components of EvaluatingTrust

ConsumingTrust contains the computing components for the control and management of trust consuming. Consuming trust deals with how to use the output of the evaluation of a trust relationship. Beyond the immediate consumption

by requesting users/applications, the evaluation result of trust relationship can be stored and/or distributed in differ ent ways. It is possible to generate credentials based on th evaluation result of a trust relationship for further distribu tion and usage. The evaluation result of a trust relationshi can also be stored in database that can be retrieved in th future. It is also possible for the evaluation result to be con sumed by the trust management system itself or by an audit ing system. In ConsumingTrust (see Figure 5), "Consumin Controller" is the manager of trust consuming. It receive the result of trust evaluation and assigns consuming task to "Direct Trust Consuming Controller", "Credential Gen erator Consuming", and "Data Storage Consuming", an "System Consuming". These components should be the im plementation of corresponding generic components of gen eral trust management architecture for distributed informa tion systems [4]. For "Direct Trust Consuming Controller and "Credential Generator Consuming", it is possible to re- quire "WS-Trust Message Handler" and "Context Informa- tion Handler" for cooperation in producing/delivering mes- sages to some involved parties.

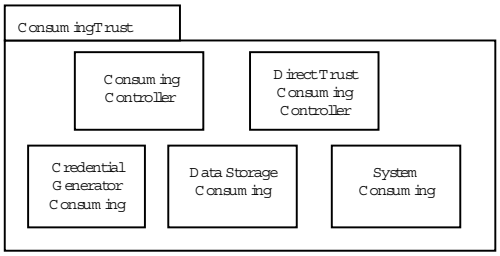

Figure 5. Components of ConsumingTrust

5. Web Services Architecture Layers

In web services paradigm, multiple web services standards and specifications have been published and adopted, in particular, there has been a stack of web services security specifications [1]. The proposed trust management layer for web services is on the top of the stack of web services security specifications. Web services security specifications can be employed to provide modular support to the trust management layer for web services. On the top of the trust management layer, it is the web services coordination layer. The trust management layer will serve WS-Coordination and WS-Transaction layers. This leads to the web services architecture layers as shown in Figure 6.

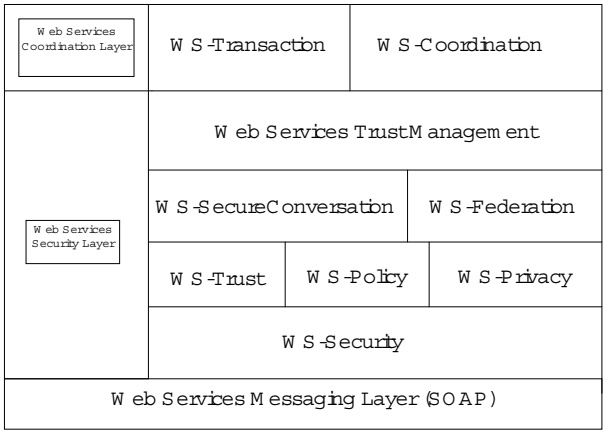

Figure 6. Web Services Architecture Layers

6. Concluding Remarks

This paper presents a comprehensive trust management approach for web services that deals with the analysis/modelling of trust relationships and the development of trust management layer in a consistent manner. The specific characteristics of trust relationships in web services have been discussed and a trust management architecture for web services have been proposed. The trust management layer in web services extends WS-Trust and covers existing trust mechanisms including credentials, reputation, data storage, and environment parameters. The results in this paper aim at providing high level guidelines, rather than a panacea, for the development of trust management solution in web services. It is hoped that the research results described in this paper can provide a solid starting point and useful high level tools for the development of the trust management solution in web services paradigm.

References

[1] I. Corporation and M. Corporation. Security in a web services world: A proposed architecture and roadmap, 2002.

[2] R. Kraft. Designing a distributed access control processor for network services on the web. In *2002 ACM workshop on XML security*, pages 36–52, Fairfax, VA, USA, 2002.

[3] W. Zhao, V. Varadharajan, and G. Bryan. Modeling trust relationships in distributed environments. *Lecture Notes in Computer Science*, 3184:40–49, 2004.

[4] W. Zhao, V. Varadharajan, and G. Bryan. A unified framework for trust management. In *2nd IEEE SECURECOMM SECOVAL Workshop: The Value of Security through Collaboration*, 2006.

[5] W. Zhao, V. Varadharajan, and G. Bryan. A unified taxonomy framework of trust. In *Trust in E-Service: Technologies, Practices and Challenges*, pages 29–50. IGI Global, 2007.