



Seamless integration of dependability and security concepts in SOA: A feedback control system based framework and taxonomy[☆]

J. Hu^{a,*}, I. Khalil^b, S. Han^c, A. Mahmood^b

^a School of Engineering and Information Technology, University of New South Wales at the Australian Defence Force Academy (UNSW@ADFA), Canberra ACT 2600, Australia

^b School of Computer Science and IT, RMIT University, Melbourne, Victoria 3001, Australia

^c DEBI Institute, Curtin University of Technology, Perth 6845, Australia

ARTICLE INFO

Article history:

Received 26 May 2010

Received in revised form

22 September 2010

Accepted 29 November 2010

Available online 7 December 2010

Keywords:

Dependability

Security

Taxonomy

SOA

ABSTRACT

Recent research effort has been made to integrate both dependability and security concepts for SOA using fault taxonomy. However most of such work is confined to the SOA functionality layer excluding the interactions with its underlying distributed systems. Also many elements of taxonomies proposed are loosely integrated without generic interactive relationships. This is especially true when security attributes are included. There is a lack of framework that can systematically and genuinely integrate dependability and security concepts for SOA and also include underlying distributed systems of SOA. This paper attempts to address this issue by providing a taxonomy and framework from a new angle. The major contribution of this paper is that we have introduced a feedback control system as an integration vehicle to integrate concepts and attributes of both dependability and security in SOA, so that they can be more generically integrated and more systematically constructed. Furthermore, the framework proposed in this paper covers the SOA functionality layer and its underlying distributed systems. A novel idea of basic fault building blocks has been proposed to address the scalability issue due to layer interactions. Various fault taxonomies are constructed from these basic building blocks.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

A service-oriented architecture (SOA) is an architecture upon which different standalone services can be loosely coupled over distributed systems. These services are virtually unrelated functional units spread over interconnected networks. Each unit provides one functional service such as filling one online form. These services can communicate with each other even though each service has a different underlying operating system, a different programming language, etc. Given a business application/task such as airline ticket booking, a composition process can find and combine relevant services so that the job can be done. New services can also be established on the fly by these existing services (Newcomer and Lomow, 2005). Fig. 1 illustrates the operational architecture of SOA and web service implementation. For convenience, we will call the architecture shown in Fig. 1 as SOA architecture unless stated otherwise. There are three layers in SOA: (i) base layer—supporting distributed systems including

networks, software, operating systems, etc.; (ii) SOA layer—SOA specific functionalities. Among them, a service provider publishes its description of service and interface information to the service registry. A service requester will find relevant service from this registry and binds to it to invoke the service; (iii) implementation layer—it provides web service applications to implement the SOA. It uses the Web Service Description Language (WSDL) for describing services; a mechanism called Universal Description Discovery and Integration (UDDI) for service registry and service discovery; and the simple object access protocol (SOAP) for exchange of messages. Web services use XML-based standards for format. The SOA layer or its implementation layer can also project itself onto an operational layer, which consists of workflow components of publishing, discovery, composition, binding and execution.

In this paper, the focus is placed on SOA. As the SOA layer is built on the base layer, it will be both meaningful and helpful to include interactions between these two layers. There are very few literature reports on fault taxonomy related to SOA (Brüning et al., 2007; Berghe et al., 2005; Montagut and Molva, 2008; Vorobiev and Han, 2006; Savolainen et al., 2007; Gudgin, 2004; Little, 2003; Tai et al., 2004; Bhiri et al., 2005; Fauvet et al., 2005). Unfortunately none has addressed such layer interactions. The SOA functionality layer sits on top of the base layer whose network platform is inherently insecure and unreliable. With ever accelerating trend of integrating mobile and wireless network infrastructure, things become worse.

[☆]This work is supported by ARC (Australia Research Council) Project ARC Discovery Grant with Project ID DP0985838, ARC Linkage Projects LP110100602, LP100200538 and LP100100404.

* Corresponding author.

E-mail addresses: j.hu@adfa.edu.au (J. Hu), Ibrahimk@cs.rmit.edu.au (I. Khalil), abdun@cs.rmit.edu.au (S. Han), song.han@cbs.curtin.edu.au (A. Mahmood).