Chapter 47
Privacy-Preserving Orchestrated Web Service Composition with Untrusted Brokers

Imen Khabou
CES Laboratory, Sfax University, Sfax, Tunisia

Mohsen Rouached
Sultan Qaboos University, Muscat, Oman

Alexandre Viejo
Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Tarragona, Spain

David Sánchez
Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Tarragona, Spain

ABSTRACT

This article describes how by using web service composition to model different business processes is a usual tendency in the industry. More specifically, web service composition enables to separate a certain process in different activities that must be executed following a certain order. Each activity has its own set of inputs and outputs and is executed by a certain web service hosted by a service provider which can be completely independent. Among all the applications in which web service composition may be applied, this article focuses on a cloud-based scenario in which a business wishes to outsource the execution of a certain complex service in exchange for some economical compensation. It is for this reason, among the different composition approaches that exist in the literature, this article focuses on the orchestrated one, in which a broker coordinates the composition. One of the main issues of orchestrated systems is the fact that the broker receives and learns all the input data needed to perform the requested complex service. This behavior may represent a serious privacy problem depending on the nature of the business process to be executed. In this article, a new privacy-preserving orchestrated Web service composition system based on a symmetric searchable encryption primitive is proposed. The main target of this new scheme is to protect the privacy of the business that wish to outsource their operations using a cloud-based solution in which the broker is honest but curious, this is, this entity tries to analyze data and message flows in order to learn all the possible sensitive information from the rest of participants in the system.

DOI: 10.4018/978-1-7998-8954-0.ch047

1. INTRODUCTION

As defined by the World Wide Web Consortium (W3C[1]), a Web Service is a software system designed to support interoperable machine-to-machine interaction over a network. In a more practical way, Web services are a standardized way of integrating Web-based applications over the Internet using open protocols that allow them to be advertised, located and composed. Precisely, composing sets of Web services to build more complex services is the raison d'être of this technology, making it a usual choice when designing and deploying Service Oriented Architectures (SOA) and Cloud Computing solutions. Composing different Web services to perform a complex sequence of activities is a natural way of modeling a business process in which different operations, with certain inputs and outputs must be executed following a specific sequence. The use of this approach for implementing business processes has received significant attention in both the scientific community and the industry.

Generally, researchers classify the techniques for Web service composition into two main classes: service choreography and service orchestration.

The choreographed scenario is based on a fully decentralized model in which Web services organize themselves in a dynamic way; in the orchestrated approach, there is a central entity, named broker, which manages the sequence of Web services that have to be executed and distributes the required inputs among them (Carminati et al., 2015). Both approaches have advantages and shortcomings: the distributed design of the service choreography is a significant asset against the availability and resiliency problems that orchestration based methods are prone to suffer due to their centralized nature; on the other hand, the lack of a broker that organizes the sequence of activities to be performed, forces choreographed proposals to assume that all the Web services to be involved in a certain execution know the existence and the interfaces of the others. This assumption is quite unrealistic in scenarios in which Web services are expected to be hosted by external entities and to be ideally applied to a wide range of different applications requested by a wide range of different users.

The scenario considered in this paper is based on offering a Cloud-based solution in which a user (ideally, a business) wishes to outsource the execution of a certain complex service; asks the broker of the system to organize the requested computation; and pays the corresponding charges. The broker will organize the distributed execution of the service among the different Web services hosted by a set of independent Service Providers that may receive some economical compensation too.

Due to the fact that the brokered service orchestration is the best suited approach to deal with the envisaged scenario, the rest of the paper will focus on it.

A main shortcoming which is inherent to brokered systems is the fact that the system's broker receives and learns all the input data needed to perform the requested complex service. This behavior may represent a serious privacy problem depending on the nature of the business process to be executed, more specifically, applications focusing on healthcare or e-commerce may involve very sensitive information from the individuals and should receive special attention.

In general, previous proposals for the service orchestration model deal with privacy issues simply considering the broker as a fully-trusted entity that follows a defined protocol and respects the privacy of the users by applying some basic privacy rules or policies that try to assure individuals that their data will be handled in a trustworthy manner (Xu et al., 2006; Tbahriti et al., 2011; Nyre et al., 2011). On the one hand, assuming that brokers are fully-trusted is such a strong assumption which may be considered unrealistic. On the other hand, one of the challenges of this environment is providing the users a framework for defining and enforcing fine-grained privacy policies that allow them to model