A policy-based evaluation framework for Quality and Security in Service Oriented Architectures

V. Casola, A. R. Fasolino, N. Mazzocca, P. Tramontana
Dipartimento di Informatica e Sistemistica
Universita' degli Studi di Napoli, Federico II. Naples, Italy
{casolav, fasolino, n.mazzocca, ptramont}@unina.it

Abstract

In dynamic cooperative architectures that are based on services (SOA), Customers are not only interested in service functionalities, but also in their quality, such as performance, cost, reliability, security and so on. In this scenario, models, techniques and tools supporting the selection of the best service are needed. In this paper, we propose an evaluation framework that includes a flexible quality meta-model for formalising Customer and Provider views of quality, and a decisional model defining a systematic approach for comparing offered and requested quality of services. We also illustrate the applicability of the framework in a Web Service(WS) scenario.

1. Introduction

Service oriented Architecture (SOA) is the emerging architectural model to build cooperative infrastructures based on loosely coupled interactive software agents. Cooperative services are capable of intelligent interaction and are able to discover and negotiate with each other, mediate on behalf of their users and compose themselves into more complex services. In this context, the problem of guaranteeing a given "quality" of services to final users, in terms of functional and non-functional requirements, like performance or security, is one of the hot topics. In general, a service provider is able to guarantee a predefined service level and a certain security level but, at the state of the art there is no way to automatically measure it. Usually, this problem is faced by an explicit agreement among services specified in the so called "Service Level Agreement" (SLA for short). At the state of the art, SLAs are primary related to the quality of service and not to security, although some security metrics have been proposed to classify, evaluate and compare security practices [3] and some recent works have also introduced the concept of "quality of protection" [1] to intend all those security and trust attributes that need to be addressed in the SLAs. To reach the aim of SLA dynamic management to support the interoperability among services, a formalized approach both for defining the different quality factors of a service in a SLA, and for evaluating them automatically is needed.

Current WS specification on security, trust and agreements (WS-Security, WS-Trust) promote the adoption of policies as the basis on which to interoperate. Moreover, policy languages are now available (WS-Policy, WS-Agreement, and so on) but they do not specify how to automatically evaluate and compare the related security and quality provisions. These problems can be approached using the two-component framework proposed in the following section.

2. The framework for SLA definition and evaluation

The first component of the framework consists of a SLA policy meta-model providing a flexible approach to define quality criteria and formalize them in an unambiguous way by policies. As Figure 1 shows, the SLA policy meta-model supports a hierarchical view of quality that decomposes higher level quality characteristic into lower level measurable ones. Unambiguous definitions of Quality Model elements can be also obtained by adopting semantic descriptions provided by ontologies, such as in [4].

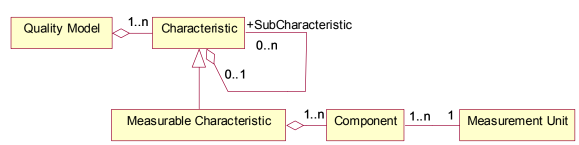

Figure 1: Quality Meta-Model

The policy meta-model can be instantiated by Customers or Providers in order to define the specific Quality Models on which their policies will be based. In the context of SLA dynamic management, the selection of a service will require a decision making process to discriminate among Providers' quality offers and specified Customer quality request.

To carry out this process, we propose to adopt the decisional technique defined by the Analytic Hierarchy Process (AHP) proposed by Thomas L. Saaty [5, 6]. This technique separates the decision framework design activity from the decision making one. The design activity is preliminarily performed by the entity (e.g., Customer or Third party) that is responsible for defining the evaluation criteria, and can be carried out with a three-step approach: in the first step (*Weight Assignment* Step), the relative importance of any pair of quality characteristics listed in the Request Model is rated, while in the remaining two steps (named *Clustering* and *Rating*) the problem of transforming the (not-homogeneous) measures of all characteristics into one standard unit of measurement is addressed. The decision making activity will be performed any time a comparison between requested and offered policies is needed. The decision will rely on an additive scoring method, where the Overall Satisfaction of a quality model will be obtained by aggregating the evaluations of all characteristics of the Quality Model, which are in turn evaluated by aggregating the evaluations of their Sub-Characteristics. The evaluation of each measurable Characteristic is computed through a corresponding Satisfaction Function (that depends on the Rating values designed in the decision framework design activity). Additional details about the framework can be found in [2].

3. Application of the Evaluation Model

The proposed evaluation framework has been used in a case study to solve the problem described in Figure 2 where a Customer, searching for a service with desired SLA, expresses a Quality Request policy and compares it against available provider's Quality Offers; we suppose that both Customer and Providers adopt the same Quality Model including the same set of Characteristics and associated Metrics to instantiate their policies.

For brevity's sake, in Table 1 we just report some policy Characteristics (e.g., *Response Time*, a *Time Behavior* characteristic, and *Confidentiality* that is a *Security* characteristic), their measurable Sub-Characteristics with the corresponding weights, and the associated Satisfaction Functions (S) of two providers' offers.

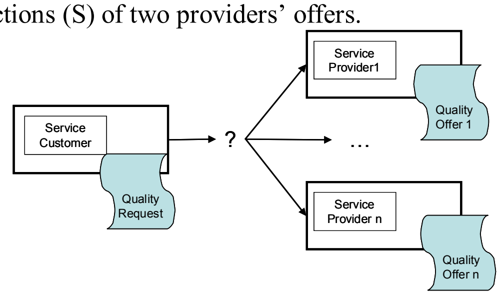

Figure 2: The problem of finding a provider

The Satisfaction evaluation of each Characteristic of a quality offer will be obtained as a weighted sum of the Satisfaction (S) rate of its Sub-Characteristics by using the weights indicated in parenthesis. As an example, the rating of *Response Time Satisfaction* and *Confidentiality* of the first offer are computed as follows:

$S_{ResponseTime}$(Customer,Provider1) = 0.64*0.26+0.07*0.75+ +0.28*0.07=0.24;

$S_{Confidentiality}$(Customer,Provider1)= 0.12*0.8+0.12*0.88+ +0.38*0.75+0.38*0.19=0.56;

The analogous evaluations of the second offer are:

$S_{ResponseTime}$(Customer,Provider2)= 0.16;

$S_{Confidentiality}$(Customer, Provider2)= 0.50

Table 1: Characteristics in Customer's and Providers' Policies

Characteristic Name	Sub-Characteristic	Customer's Values	Provider1's Values	Provider2's Values
Response Time	Average RT (0.64)	1.5 s	1 s S= 0.26	1.6 S= 0.11
	Max RT (0.07)	2 s	1.7 s S= 0.75	2 s S= 0.25
	Maximum RT (0.28)	0.2 s	0.4 s S= 0.07	0.2 s S= 0.25
Confidentiality	Alg (0.12)	RSA	RSA S=0.8	RSA S=0.8
	MessagePart (0.12)	Body	Body S= 0.88	Body S= 0.88
	KeyLen (0.38)	512 bit	1024 bit S= 0.75	512 bit S= 0.19
	KeyLoc (0.38)	HD	Smart Card S= 0.75	Floppy S=0.06

The overall Satisfaction Degrees of both policies are obtained by aggregating the characteristics' evaluations:

$$S(Cust, Provider1) = \sum_{c \in C} w_c S_c (Cust, Provider1) = 0.43$$
$$S(Cust, Provider2) = \sum_{c \in C} w_c S_c (Cust, Provider2) = 0.34$$

In conclusion, the service offered by Provider 1 can be considered better than the service offered by Provider 2, since the former policy has a greater Satisfaction value (0.43 versus 0.34).

5. References

[1] V. Casola, A. Mazzeo, N. Mazzocca, M. Rak. A SLA evaluation methodology in Service Oriented Architectures. Proc. of Quality of Protection Workshop 2005, Springer-Verlag.

[2] V. Casola, A. R. Fasolino, N. Mazzocca, P. Tramontana. A framework for evaluating Quality and Security in Service Oriented Architectures. http://www.seclab.unina.it

[3] DoD (Department of Defense). Trusted Computer System Evaluation Criteria, 26 December 1985 (Orange Book).

[4] N. Oldham, K. Verma, A Sheth, F. Hakimpour. Semantic WS-Agreement Partner Selection. Proc. of the 15th International Conference on World Wide Web, WWW '06. ACM Press.

[5] T.L. Saaty. How to make a decision: the analytic hierarchy process, European Journal of Operational Research, 1990. Vol. 48, n.1, Elsevier, pp. 9-26.

[6] L. Zhu, A. Aurum, I. Gorton and R. Jeffery, Tradeoff and Sensitivity Analysis in Software Architecture Evaluation Using Analytic Hierarchy Process, Software Quality Journal, Springer Netherlands, Volume 13, Number 4 / December, 2005