An AHP-based Framework for Quality and Security Evaluation

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

Rationale

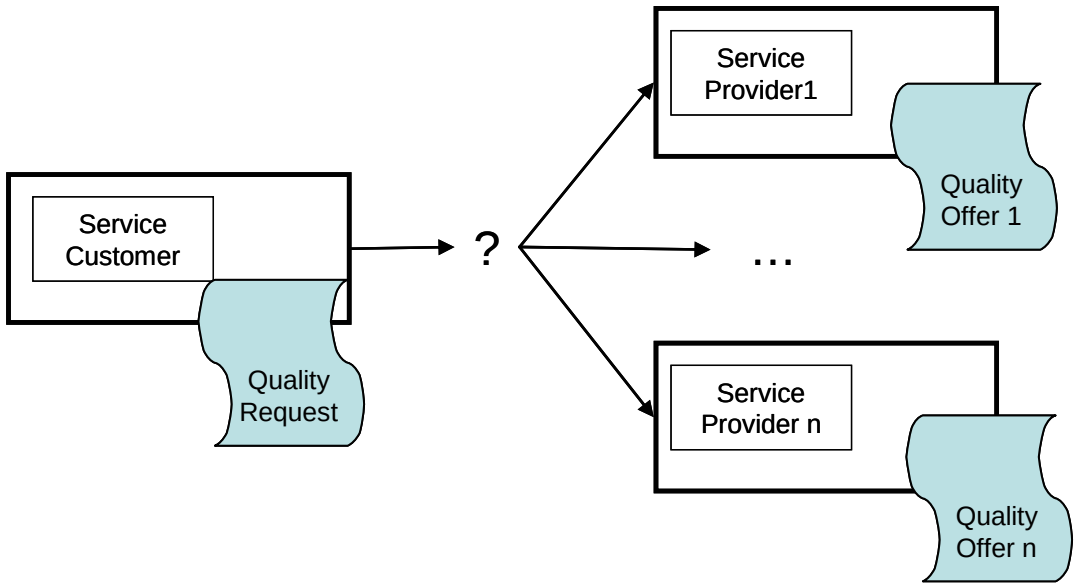
- Context and open issues
- Our approach to quality and security evaluation
- Methodology
  - Policy Formalization
  - Evaluation technique
- Applicability in Web Service Architectures
- Conclusions and Future Works

Context: service cooperation, a security point of view

- Service Oriented Architectures are capable of intelligent interaction and are able to discover and compose themselves into more complex services;

- The open issue is: how to guarantee the “quality and security” of a service built at run-time in a potential un-trusted domain?

How a Customer can choose the Web Service that better fits his “quality” requirements?



Our approach to quality/security evaluation

- Actually, these problems are faced by explicit agreements among services:
  - Each service defines its own **Service Level Agreement** and **Security Policy** and publishes them in a public document;
  - People from the various organization that want to cooperate, manually evaluate the different SLAs and decide to agree or not.
- SLA and Policies are expressed by means of a free text document; they usually contain provisions on the “quality” of services and on “security” mechanisms adopted, they are used to decide to extend trust to other services;
- These documents are mostly manually evaluated.

Security evaluation Methodology

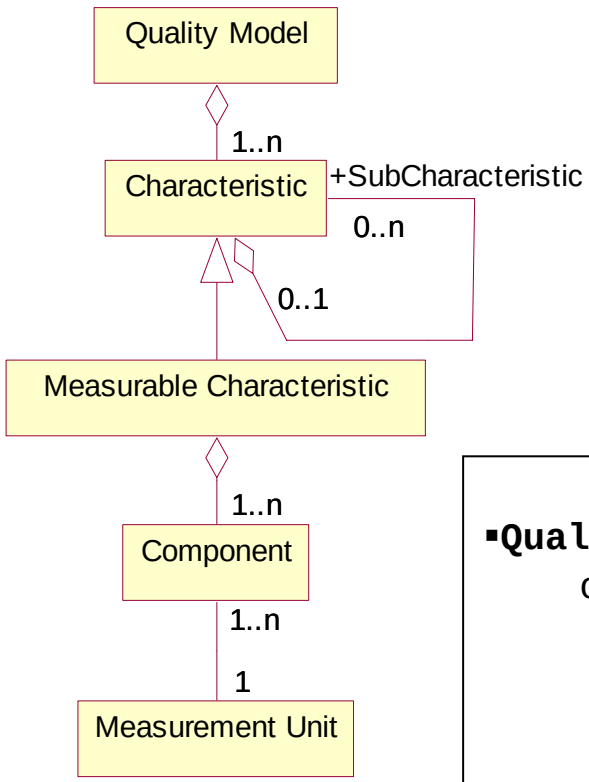
- We are working on different methodologies to:
  - **Express quality/security** through a semi-formal and not ambiguous model; the chosen formalization must be “easy to adopt” for technical and organizational people;
  - **Evaluate the quality/security level** that a security infrastructure is able to guarantee by aggregating the security associated to all policy provisions (multidecision approach).
  - **Compare** different services according to the measured quality/security level.

The proposed approach

- Models are needed to formally express the **Quality and security** of Web Services (quality of protection, QoS, security and so on) requested by Customers and offered by Providers;

1. We defined a quality meta-model and formally express Quality as an instance of the meta-model;
2. We investigated the adoption of a decision framework based on AHP (Analytic Hierarchy Process) proposed by Saaty for Quality evaluation;

The Quality Meta-Model



• **Quality Characteristic**: any quality requirements, such as Performance, Security, Cost, Maintainability

•**Characteristics** may be arranged in a hierarchy (Measurable Characteristics are the leaves)

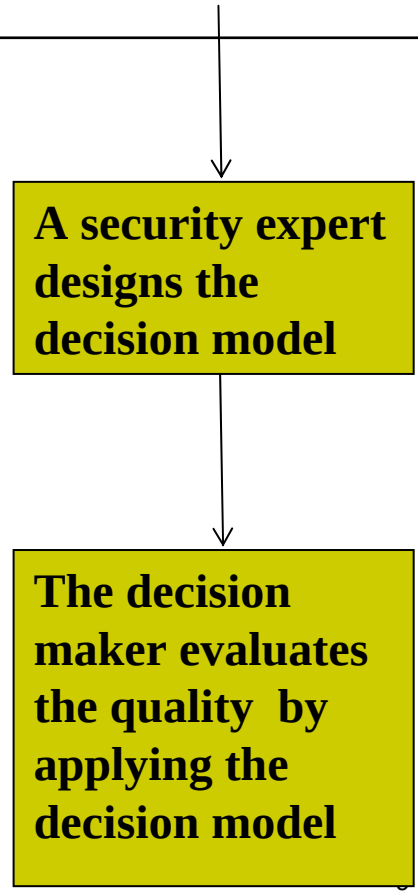
• **Measurable Characteristic**: a Quality Characteristic that can directly be measured

- **Quality Characteristic: Efficiency**
  - **Quality Characteristic: Time Behavior**
    - **Quality Characteristic:** Response Time
      - **Measurable Quality Characteristic:** Average Response Time
      - **Measurable Quality Characteristic:** Standard deviation
      - **Measurable Quality Characteristic:** Maximum response time

The Analytical Hierarchy Process

1. **The decision model design activity:**
   1. **Weight Assignment step**: the relative importance of the characteristics is rated;
   2. **Clustering step:** for each measurable characteristic, the sets of values that will be considered equivalent for the aims of the evaluation are defined;
   3. **Rating Step:** each set is associated to a rating value;

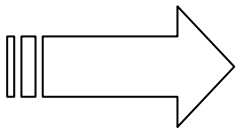
2. **The decision making activity:** to compare the quality of an offered service (formalised in a Quality Offer Model) against requestor needs (formalised in a Quality Request Model)



Building the decisional model: Step 1: Weight Assignment

For each Characteristic that is not directly measurable,
the decision process designer will estimate the relative **Intensity of Importance** of any pair of its *n* Sub-Characteristics, by defining a matrix of *nxn*

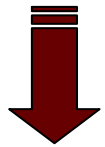
Intensity of Importance and its interpretation	
Intensity of Importance	**Interpretation**
1	Equal Importance
3	Moderate Importance
5	Strong Importance
7	Very strong Importance
9	Extreme Importance



1. Build the Comparison matrix

Response Time	Average Response Time	Standard Deviation Response Time	Maximum of Response Time
Average Response Time	1	3	7
Standard Deviation Response Time	1/3	1	5
Maximum Response Time	1/7	1/5	1

2. Normalize The matrix



$$m(i,j) = 1/m(j,i) \ \forall i,j$$
$$m(i,i) = 1 \ \forall i$$

Building the decisional model: Step 1: Weight Assignment (cont.)

2. Normalize The matrix

$$m'(i,j) = m(i,j) \Big/ \sum_{h=1}^{n} m(h,j) \quad \forall \text{i,j}$$

Characteristic Weights are assigned by comparing their relative importance:

$$w(i) = \sum_{k=1}^{n} \frac{m'(i,k)}{n} \quad \forall \text{i}$$

	Average Response Time	Standard Deviation Response Time	Maximum Response Time	Weights
Average Response Time	21/31	15/21	7/13	0.64
Standard Deviation Response Time	7/31	5/21	5/13	0.28
Maximum Response Time	3/31	1/21	1/13	0.07

Building the decisional model: Step 2: Clustering

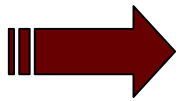
We need an Utility Function to ORDER the possible values on the basis of relative (and not absolute) preferences (LOCAL SECURITY LEVELS).

In general, an Utility function R assigns ordered values (of utility) to members of a set: given two values x and y of the set, if x is preferred to y then R(x)> R(y).

Example: Average Response Time characteristic

R = Offered_value / Requested_value

Possible Solutions are clustered in three levels:



$R < 0.5$ (very fast response);
$0.5 \leq R < 1$ (sufficiently fast response);
$1 \leq R < 2$ (quite slow response).

Building the decisional model: Step 3: Rating

After clustering each possible value, we need to rate such clusters according their **goodness**

Intensity of Goodness and its interpretation	
Intensity of Goodness	**Interpretation**
1	Equivalent
3	Moderately better
5	Strongly better
7	Very strongly better
9	Extremely better

Ratings are assigned to clusters by comparing their relative Goodness

	R<0.5	0.5≤R<1	1≤R<2	Rating
R < 0.5	1	3	5	0.63
0.5≤R<1	1/3	1	3	0.26
1≤ R<2	1/5	1/3	1	0.11

Satisfaction Function $S_{sc}(R)$

$$S(R) = \begin{cases} 0.63 & if \quad R < 0.5 \\ 0.26 & if \quad 0.5 \le R < 1 \\ 0.11 & if \quad 1 \le R < 2 \end{cases}$$

This is the relative rate/evaluation of a cluster

The Decision Making Activity

The Quality of different Web Services is compared by evaluating:

1. **a Satisfaction Function for each Measurable Characteristic.**
2. **a Satisfaction Function for each non-Measurable Characteristic:**

$$S_c(request, offer) = \sum_{sc \in C(c)} w_{sc} S_{sc}(request, offer)$$

A non measurable characteristic (c), For example: Confidentiality

All measurable sub-characteristic of (c) denoted sc are weighted and summed

For example: (Encryption Alghoritm, KeyLenght, KeyProtection,)

The Decision Making Activity (cont.)

3. the Overall Satisfaction Function:

$$S(request, offer) = \sum_{c \in Characteristic} w_c S_c(request, offer)$$

The Web Service with the greater Satisfaction Function value is chosen

Application of the evaluation model: evaluating measurable and not-measurable characteristics

Characteristic Name	Sub-Characteristic	Customer's Values	Provider1 's Values	Provider2 's Values
Integrity (0.35)	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
Response Time (0.67)	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

$S_{Integrity}$(Customer,Provider1)=
0.12*0.8+0.12*0.88+0.38*0.75+0.38*0.75=0.77

$S_{Integrity}$(Customer,Provider2)=
0.12*0.8+0.12*0.88+0.38*0.19+0.38*0.06=0.30

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

$S_{ResponseTime}$(Customer,Provider2)=
0.64*0.11+0.07*0.25+0.28*0.25=**0.16**

Application of the evaluation model: Overall evaluation

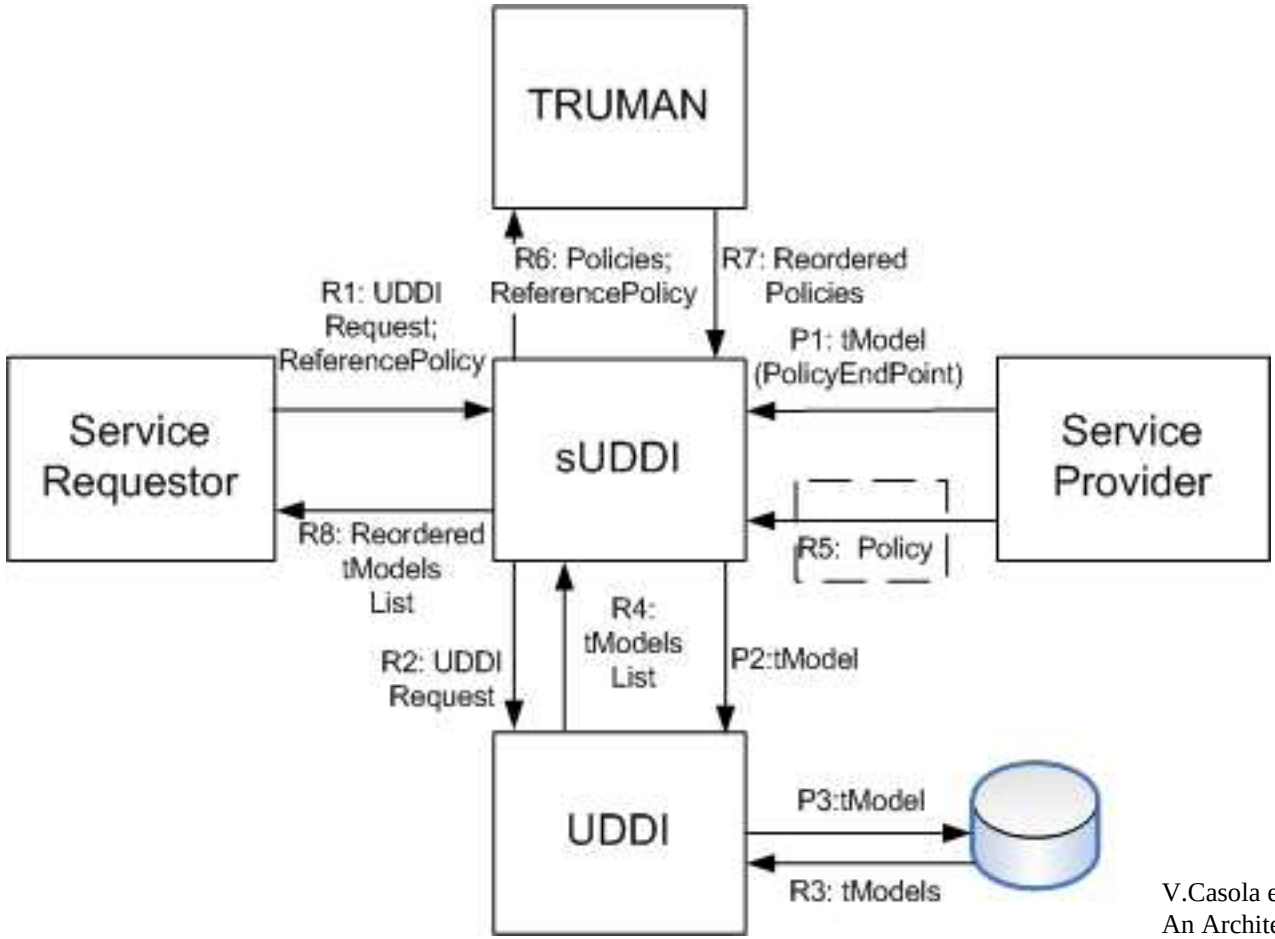
Finally we evaluate the overall satisfaction function (GLOBAL SECURITY LEVEL)

$$\sum_{c \in C} w_c S_c(Cust, \mathrm{Pr}ovider1) = 0.43$$

$$\sum_{c \in C} w_c S_c(Cust, \mathrm{Pr}ovider2) = 0.34$$

The first provider will be chosen on the basis of the provided security level

An idea on how to automatically enforce the evaluation: A reference architecture



V.Casola et al.
An Architectural Model for Trusted Domains in Web Services
Journal of Information Assurance and Security 2 (2006)

Conclusions and Future work

- We are working on methodologies to automatically evaluate quality and security provided by an internet service on the basis of the published policies;
- The AHP methodology allows to address measurable and not-measurable quality and security aspects in a unifying way and propose an evaluation model;
- We are going to integrate such methodology in the TRUMAN architecture and compare with existing ones.