A User-Centric Approach to Assessing Confidentiality and Integrity of Service-Based Workflows*

Stephen S. Yau and Jing Huang

Information Assurance Center, and
School of Computing, Informatics, and Decision System Engineering
Arizona State University, Tempe, AZ 85287-8809, USA
{yau, Jing.Huang.5}@asu.edu

***Abstract*— In service-based systems (SBS), workflows are composed of loosely coupled services. The quality of service (QoS) of composite service-based workflows needs to be considered. In doing so, various aspects of the QoS of these workflows need to be assessed. In this paper, a user-centric approach is presented to assessing the confidentiality and integrity of service-based workflows, given the QoS of constituent services. Our approach incorporates the users' preferences for data sensitivity on the data processed by the workflow. It allows users to compare the amount of risk incurred by possible failure of the confidentiality or integrity functions of component services of a workflow.**

Keywords - service-based workflow; user preference; security; confidentiality; integrity; multiple QoS

I. INTRODUCTION

Applications in service-based systems (SBS) are developed in an efficient, low-cost and flexible way based on composing loosely coupled services to form the workflows for the applications. To do so, a service needs to be found to realize the functionality of each of the tasks in the workflow. Using semantic matching, there may be more than one candidate service to fulfill the functional requirements of a task. Managing the non-functional quality of service (QoS) is challenging for workflow management, and approaches, such as adaptive resource allocation [1] and tradeoff between performance and security [2], have been presented to ensure that multiple workflows in a SBS are allocated with sufficient system resources satisfying their QoS requirements. The ability of identifying the QoS of workflows facilitates the users to conduct QoS-based design, QoS-based selection and execution, QoS monitoring and QoS-based adaptation of the workflows [3]. In order to achieve this capability, we need an effective approach to assess various QoS aspects of workflows, including performance, throughput, accuracy, delay and security. The assessment of various aspects of QoS of service-based workflows have been widely studied [3, 4, 5-8], but the assessment of one of the most important QoS aspects, the security, is missing.

*This work reported here is sponsored by National Science Foundation under Grant No. CCF-0725340.

In this paper, we will present an approach to assessing two of the most important aspects of security for service-based workflows: confidentiality and integrity. Our assessment of confidentiality and integrity of service-based workflows will incorporate the preferences of the users for data sensitivity. Our approach can provide the users with the information on the amount of risk on their assets due to the possible failures of the confidentiality or integrity of component services of the workflow.

This paper is organized as follows: In Section II we will discuss the current state of the art on assessing the security of service-based workflows, including confidentiality and integrity. An overview of our user-centric approach to assessing the confidentiality and integrity of service-based workflows will be presented in Section III. We will present our approach to assessing the confidentiality and integrity of service-based workflows in detail with illustrative examples in Sections IV and V, respectively. In Section VI, we will discuss our approach and the directions of future research in this area.

II. RELATED WORK

So far, there is no commonly acceptable measure for the security for a complex software system, and no systematic approach to conducting practical security evaluation of software-intensive systems [9]. Savola [10] introduced a holistic framework for security evaluation based on security behavior modeling and security evidence collection. In [11], a security measure, called the *mean time* (or *effort*) *to security failure* and a method of computing the probabilities of security failure due to violations of various security attributes were presented. Abedin et al. [12] introduced a quantitative method to evaluate the quality of protection of security policies by combining historical vulnerability, current vulnerability, traffic volume and network exposure, such as the number of IP addresses and ports. For software design and construction, the security of a pattern-based software architecture can be assessed by aggregating a set of security objectives and threats to determine the overall protection [13]. A quantitative method to evaluate security services was presented in [14] using a normalized weighted tree. Artaiam and Senivongse [15] used CVSS scores [16] of the underlying system to compute the security value of a service. Hwang et al. [17] presented a

978-1-4244-7570-4/10/$26.00 ©2010 IEEE

unified probabilistic model to model QoS values including response time, reliability, fidelity rating, and cost. We have developed a method for evaluating probabilistic security of services using cryptographic settings [2]. Butler [18] applied risk analysis to security assessment to help the stakeholders select the most cost effective security alternatives.

Workflow control patterns were introduced by van der Aalst, et al [19] to identify comprehensive workflow functionality. Russell et al. [20] used workflow data patterns to characterize the data activities in a workflow by data visibility, data interaction, data transfer and data-based routing. Among them, data interaction refers to the manner in which data is transferred between elements within a workflow, and data-based routing examines the manner in which data elements can influence the operations of other aspects of the workflow, especially the control flow perspective. They will be considered relevant factors in our approach

Cardoso et al. [3] presented a model to compute the QoS for workflows based on atomic task QoS attributes. This model includes three QoS dimensions: time, cost and reliability, and was implemented for the METEOR workflow system. Menasce [5] and Jaeger et al. [4] studied QoS-aware Web services composition, where QoS attributes are aggregated according to workflow composition patterns [18].

Jaeger, el al. [6] extended the composition patterns to consider dependencies among services in the same domain. In the case where given availability statements do not cover individual services, but refer to an entire server or to the connection to a network, a dependency domain is formed and represented by a single QoS statement that covers a number of services. They also showed how the information gained from monitoring process can help calculate a more accurate aggregation of QoS.

Gonczy el al. [7] applied dependability analysis of Multiple Phased Systems (MPS) [21] to dependability evaluation of business processes. Yang, el al. [8] presented the process of QoS-aware service discovery and the absolute and relative matchmaking criteria, and represented confidentiality, integrity and authentication among other QoS attributes using security tokens for estimation, but did not give the aggregated effects of these attributes. Massacci and Yautsiukhin [22] developed an overall assurance indicator of a complex business process from its components for arbitrary monotone composition functions. Their assurance indicator is a measurable indicator negotiated by a client and a contractor to show that the client's business assurance goals are addressed, such as the number of attacks or breaches affecting the clients' assets.

III. Overview of Our User-Centric Approach

Quality assessment for a workflow is based on the estimated quality of component services. For example, the throughput of a service is the number of requests the service can respond in a given period of time. The throughput of a workflow is the number of requests the workflow can process per unit time and it depends on the throughput of all the constituent services of the workflow as well as the workflow structure. Similarly, the security quality of a workflow is based on the security qualities of the constituent services of the workflow.

Before we discuss our overall approach, we need to define the confidentiality and integrity estimates of services and workflows.

A. Confidentiality and Integrity Estimates of Services

In this paper, the *confidentiality estimate of a service* is defined as the probability that the information is kept secret from unauthorized entities throughout the execution of an instance of the service. The i*ntegrity estimate of a service* is defined as the probability that the information produced by the execution of an instance of the service is not manipulated by unauthorized entities. By "throughout the execution", we mean from the time when the service receives its input until the time when it finishes processing the necessary data and sends the results out as its output.

B. Our User-Centric Approach

One important goal of security is to protect the information processed or flowing through the workflow. Therefore, merely evaluating the standalone system without considering various relations between the information in the system and that in the components of the system will not help the user make good decisions in selecting security alternatives. Security is always related to risk, and applying risk analysis to security assessment helps the stakeholders select the most cost effective security alternatives [18]. Therefore, we let the user identify the important data and the expected value of the data which refers to the estimated cost (damage) due to the exposure of the data to unauthorized users, and our approach will incorporate such user's expectation accordingly.

For confidentiality, we are concerned with keeping the secrecy of our sensitive information. Hence, to evaluate the confidentiality, the information we need first is what sensitive data needs to be accessed by the services in the workflow. As mentioned before, we let the user specify the sensitive data items which are considered important to be kept confidential, and their expected values. In this way, the user of the workflow is given the flexibility of adapting the assessment of confidentiality according to the importance of the data. When such values change for whatever reasons, we can re-evaluate the workflow to help the user make the best decision at the time.

Let dc_j, $1 \le j \le m$ denote the j-th sensitive data item and vc_j be the value of dc_j, the *confidentiality estimate of a workflow* W can be defined as follows:

$$E\_c(W) = 1 - \frac{\sum_1^m (1 - C(dc_j)) vc_j}{\sum_1^m vc_j} \quad (1)$$

where $C(dc_j)$ is the probability that dc_j is kept confidential during the execution of the workflow.

Integrity deals with the authenticity of information. Therefore, we focus on the data which is generated or modified as a result of the execution of the entire system. We let the user specify a list of sensitive data items and their values which refer to the estimated cost due to the manipulation of the data. Similarly, let di_j, $1 \le j \le m$, denote the j-th sensitive data item and vi_j be the value of di_j. The *integrity estimate of a workflow* W can be defined as follows:

$$E\_i(W) = 1 - \frac{\sum_1^m (1 - I(di_j))vi_j}{\sum_1^m vi_j} \quad (2)$$

where $I(di_j)$ is the probability that di_j is not manipulated during the execution of the workflow.

Our estimated confidentiality and integrity of a workflow reflect the risks associated with the sensitive data of the workflow. Our estimates are functions of not only the workflow control patterns, the confidentiality and integrity of the constituent services, but also the user's concerns regarding sensitive data, which can be the input or output, or both, of the services of the workflow. Such an assessment is adaptive to the user's input which may change over time.

In the next two sections, we will present our approach in detail to assessing the confidentiality and integrity of service-based workflows.

IV. Assessing User-Centric Estimate of Confidentiality

Reduction has been used to derive QoS for composite workflows [3] [4] by collapsing the workflow control graph into a single task. In our approach, as probabilistic values, the aggregated confidentiality or integrity of a sequence structure is the multiplication of the estimates of all the services along the sequence; the aggregated confidentiality or integrity of a XOR-split structure is the weighted sum of the components' estimates, where the weighting factors are the probabilities of executing component services. Table I summarizes the reduction rules to integrate the confidentiality and integrity estimates of component services for assessing the confidentiality and integrity of the workflow. The corresponding workflow control patterns are shown in Fig 1.

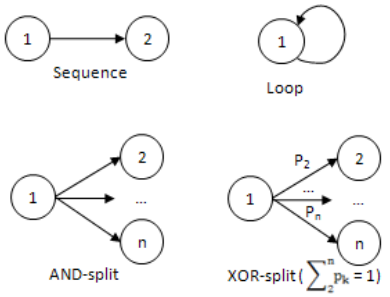

Figure 1. Basic workflow control patterns

TABLE I. Reduction Rules of Confidentiality and Integrity for Basic Control Patterns

Patterns	Aggregated Estimate
Sequence	$E_1 \times E_2$
Loop	$(E_1)^n$
AND-split	$E_1 \times \ldots \times E_n$
XOR-split	$E_k \times (\sum_2^n p_k E_k)$

Let S_i denote the candidate service selected for the i^{th} task T_i of the workflow, and C_i the estimated confidentiality of S_i. To calculate $C(dc_j)$, we are only interested in those services which have access to dc_j when they are executed, i.e., taking d_j as an input. We use $A(dc_j)$ to denote the set of such services.

Our approach to computing *E_c(W)* can be described as follows:

Step C1) Derive the aggregated confidentiality of the entire workflow expressed with C_i using the reduction rules in Table I.

Step C2) Calculate $C(dc_j)$ for each sensitive data item dc_j:

a) Identify $A(dc_j)$: the set of services that have access to dc_j during the workflow execution;

b) Calculate $C(dc_j)$ using the confidentiality expression derived in *Step C1)*. If a service S_i does not belong to $A(dc_j)$, $C_i = 1$.

Step C3) Compute *E_c(W)* using (1).

Now, we would like to use the following example to illustrate our approach to computing *E_c(W)*.

Example 1

Consider a workflow in an online sales system, which processes an order using a credit card after the order is submitted [23]. The workflow control graph is shown in Fig. 2. The tasks' descriptions are summarized in Table II. We will assess the confidentiality of the workflow.

Among the five tasks in Table II, T_2 and T_4 involve the credit card issuing bank and the merchant bank. T_5 involves some shipping company. When an order is submitted by a customer, the merchant will first confirm with the warehouse that the merchandise is in stock and hence the merchant can fulfill the order. At the same time, the merchant checks the validity of the customer's credit card. If the customer's credit card does not pass one of the checks, the order is terminated with the failure reason indicated in the notification sent to the customer. Otherwise, a request of payment is sent to the bank issuing the credit card which pays the merchant's bank. If the payment is not successful, the order is still terminated; otherwise, start the shipping process. Suppose the sensitive data items and their values are provided in Table III.

In *Step C1)*, the reduction rules in Table I are applied to the workflow, and the aggregated confidentiality of the workflow derived based on the structure of the control flow shown in Fig. 2 is:

$$C_1 \times C_2 \times (1\% \times C_3 + 99\% \times C_4 \times (0.1\% \times C_3 + 99.9\% \times C_5)) \quad (3)$$

TABLE II. TASKS OF THE WORKFLOW AND RELATED DATA OF EXAMPLE 1.

T_1	function	confirm stock status with the warehouse
	input	merchandise, amount
	output	success/fail
T_2	function	credit card check
	input	dc_1
	output	success/fail
T_3	function	terminate the order
	input	error code, order #, dc_4
	output	order record, notification email
T_4	function	process payment
	input	dc_1, dc_3
	output	success/fail, bank account records
T_5	function	process shipment
	input	dc_2, dc_4
	output	shipping record

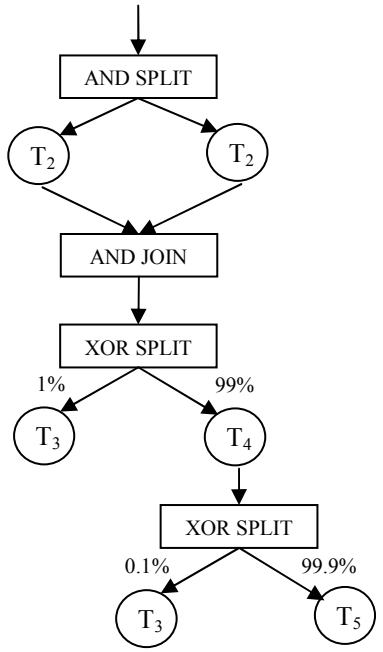

Figure 2. The workflow control graph of Example l.

TABLE III. USER SPECIFIED SENSITIVE DATA ITEMS OF EXAMPLE 1.

Data Item	Name	Contents	Value
dc_1	customer billing information	credit card #, name on card, card verification code, billing address	100
dc_2	shipping address	customer shipping address	1
dc_3	MID/TID	merchant ID and terminal ID	30
dc_4	customer contact information	customer email address, phone number, etc.	2

In *Step C2)* because the same method is used to compute $C(dc_j)$, we only present the computation of $C(dc_1)$. First, we identify the services having access to dc_1. In this case $A(dc_1) = \{S_2, S_4\}$. Suppose the estimated confidentiality of the current collection of candidate services are $\{C_1 = 0.7, C_2 = 0.98, C_3 = 0.8, C_4 = 0.99, C_5 = 0.7\}$. By substituting C_1, C_3 and C_5 with 1 to compute $C(dc_1)$ according to (3), we obtain

$$C(dc_1) = 1 \times C_2 \times (1\% \times 1 + 99\% \times C_4 \times (0.1\% \times 1 + 99.9\% \times 1)) = 97.03\%$$

Similarly, we have $C(dc_2) = 70.33\%$, $C(dc_3) = 99.01\%$, $C(dc_4) = 70.11\%$.

In *Step C3)*, we use (1) to combine $C(dc_j)$ and the values of data items. The resultant estimate is normalized within the range of [0, 1] and satisfies the following properties:

P1) When all the $C(dc_j)$ are of value 1, all the sensitive data is guaranteed to be kept confidential, and the result of (1) is 1, which is the highest assurance that can be achieved;

P2) When all the $C(dc_j)$ are of value 0, all the sensitive data is guaranteed to be exposed, and the result of (1) is 0, which is the lowest assurance that can be achieved;

P3) If we substitute a candidate service with another, the difference of their confidentiality estimates will affect the final assessment more when they have access to more valuable sensitive data.

In this example, $E\_c(W)$ is

$$1 - \frac{(1 - 97.03\%) \times 100 + (1 - 70.33\%) \times 1 + (1 - 99.01\%) \times 30 + (1 - 70.11\%) \times 2}{100 + 1 + 30 + 2} = 0.7476.$$

If we substitute S_2 and S_5 with two functionally equivalent services so that C_2 decreases from 0.98 to 0.88, while C_5 increases from 0.7 to 0.8, the final estimated $E\_c(W)$ will be 0.6754, which is less than the current one. This illustrates the above property *P3)*.

Such an estimated $E\_c(W)$ can be incorporated in the optimization process of multiple QoSs to find the optimal selection of candidate services under multidimensional user constraints.

V. ASSESSING USER-CENTRIC ESTIMATE OF INTEGRITY

As in Section IV, let S_i denote the candidate service selected for the i^{th} task T_i of the workflow, and I_i is the estimated integrity of S_i. Computing $I(di_j)$ is not as straightforward as that of $C(dc_j)$. Although di_j may be generated by a single service S_i, all the services that directly or indirectly affect the execution of S_i could affect di_j. To find all the services that have impact on di_j, we first identify the task that eventually updates di_j in the system, and then backtrack those tasks which generate intermediate results for this task.

Data produced by one task T_u affects the execution of task T_v after the execution of T_u in two possible ways. One way is that there is task level data interaction [20] between T_u and T_v, which means that T_v takes the output of T_u as its input. For example, a service that calculates the loan rate needs the result of the service which calculates the credit score of the borrower. The other way is called *data-based routing* [3], in which data elements can influence the operations of other aspects of the

workflow, especially from the control flow perspective. For example, in Fig. 2, the result of S_4 determines whether S_3 or S_5 will be invoked next. If data di_j is determined by T_v, in both ways, the integrity of di_j is influenced by both instances of T_u and T_v. All tasks that have a direct impact (like T_v) or an indirect impact (like T_u) on di_j in the workflow, connected with dependency relations among these tasks, form an information flow leading to the determination of di_j. We denote this information flow $F(di_j)$. When XOR-split is involved, $F(di_j)$ may contain those tasks that do not have either a direct or indirect impact on di_j.

Our approach to computing $E\_i(W)$ can be described as follows:

Step I1) Calculate $I(di_j)$ for each sensitive data item di_j:

a) Identify $F(di_j)$;

b) Compute $I(di_j)$ from $F(di_j)$ by reduction. If a service S_i in $F(di_j)$ does not have a direct or indirect impact on di_j, $I_i = 1$.

Step I2) Compute $E\_i(W)$ using (2).

To illustrate this, let us consider the following example.

Example 2

Considering the same workflow shown in Fig. 2, Table IV lists the sensitive data items considered for integrity assessment and their values. We will estimate the integrity of the workflow.

For *Step I1)*, because the same method is used to identify $F(di_j)$, we only present $F(di_2)$ in Fig. 3. The tasks that have an impact on di_2 are shadowed. T_4 is the task that produces the bank account records. The information provided by T_1 and T_2 determines whether T_4 is executed or not. We keep T_3 in the graph for computing the estimate although the execution of T_3 does not affect di_2 in any way. Note that in this example, $F(di_j)$ happens to have the same structure as the workflow control graph shown in Fig. 2. If the two tasks connected by a data dependency link are not connected in the workflow control flow since the result of one task could be used by another task which is not executed right next, the identification of $F(di_j)$ is not affected because we use data dependencies to identify $F(di_j)$.

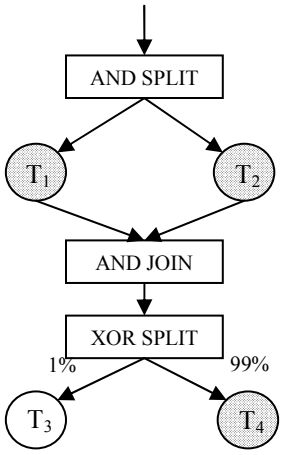

Figure 3. $F(di_2)$: information flow graph of di_2 of Example 2.

Suppose the candidate services have integrity values of $\{I_1 = 0.8, I_2 = 0.99, I_3 = 0.85, I_4 = 0.99, I_5 = 0.85\}$. Let I_3 be 1, we can compute the result of the expression derived using reduction:

$$I(di_2) = I_1 \times I_2 \times (1\% + 99\% \times I_4)$$
$$= 0.8 \times 0.99 \times (1\% \times 1 + 99\% \times 0.99) = 78.42\%$$

Similarly, $I(di_1) = 78.29\%$, $I(di_3) = 66.78\%$.

TABLE IV. USER SPECIFIED SENSITIVE DATA ITEMS OF EXAMPLE 2.

Data item	Name	Contents	Value
di_1	order record	order details, status	2
di_2	bank account records	financial transaction details	100
di_3	shipping record	shipping details	1

In *Step I2)*, $E\_i(W)$ is computed according to (2), which also satisfies the three properties discussed in Section IV. The final integrity estimate is

$$1 - \frac{(1 - 78.29\%) \times 2 + (1 - 78.42\%) \times 100 + (1 - 66.78\%) \times 1}{2 + 100 + 1}$$

=0.7905.

VI. CONCLUSION

In this paper we have presented a user-centric approach to assessing the confidentiality and integrity of a service-based workflow based on the QoS of its component services. This enables the QoS-aware workflow composition to take into account the security aspect along with other QoS aspects of a workflow. Due to the simple computation in our approach, our approach is suitable to incorporate the user's preference on data sensitivity of data processed or running through the workflow.

Future research in this area includes development of techniques for automating the security analysis of workflows to identify the relations among services and data, and further automating the entire process for assessing the confidentiality and integrity of service-based workflows. We also need to develop validation methods to validation our assessment of confidentiality and integrity.

REFERENCES

[1] Stephen S. Yau and Ho G. An, "Adaptive Resource Allocation for Service-Based Systems", Int'l Journal of Software and Informatics, Vol. 3, No. 4, Dec. 2009, pp. 483-499.

[2] Stephen S. Yau, Yin Yin and Ho G. An, "An adaptive approach to tradeoff between service performance and security in service–based systems", Proc. Int'l Conf. on Web Services (ICWS 2009), 2009, pp. 287-294.

[3] J. Cardoso, A. Sheth, J. Miller, J. Arnold, and K. Kochut, "Quality of service for workflows and web service processes", Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 1, Issue 3, April 2004, pp. 281-308.

[4] M. C. Jaeger, G. Rojec-Goldman, and G. Muehl, "QoS aggregation for Web service composition using workflow patterns," Proc. 8th Int'l IEEE

Enterprise Distributed Object Computing Conf. (EDOC 2004), Monterey, California, 2004, pp. 149-159.

[5] D. A. Menasce, "Composing Web services: a QoS view", Internet Computing, IEEE , vol.8, no.6, Nov.-Dec. 2004, pp. 88-90.

[6] Michael C. Jaeger, Gregor Rojec-Goldmann and Gero Mühl, "QoS aggregation in Web service compositions", Proc. 2005 IEEE Int'l Conf. on e-Technology, e-Commerce and e-Service (EEE'05), pp.181-185.

[7] L. Gonczy, S. Chiaradonna, F. D. Giandomenico, A. Pataricza, A. Bondavalli and T. Bartha, "Dependability evaluation of Web service-based processes", Proc. European Performance Engineering Workshop (EPEW 2006), Lecture Notes on Computer Science, In M. Telek (ed.), in. Springer, Budapest, HUNGARY, 2006, pp. 166-180.

[8] S. Yang, B. Lan and J. Chung, "Analyses of QoS-aware Web services", Proc. 2006 International Computer Symposium on Web Technologies and Information Security Workshop, 2006.

[9] Reijo Savola and Juha Röning, "Towards security evaluation based on evidence information collection and impact analysis", Proc. Int'l Conf. on Dependable Systems and Networks (DSN-2006), 2006.

[10] Reijo Savola, "Towards Security Evaluation Based on Evidence Collection", Proc. 3rd Int'l Conf. on Fuzzy Systems and Knowledge Discovery, Xi'an, China, Sep. 24-28, 2006.

[11] Bharat B. Madan, Katerina Goševa-Popstojanova, Kalyanaraman Vaidyanathan and Kishor S. Trivedi, "Modeling and quantification of security attributes of software systems", Proc. Int'l Conf. on Dependable Systems and Networks (DSN'02), 2002, pp. 505-514.

[12] M. Abedin, S. Nessa, E. Al-Shaer and L. Khan, "Vulnerability analysis for evaluating quality of protection of security policies", Proc. 2nd ACM Workshop on Quality of Protection, 2006.

[13] Artsiom Yautsiukhin, Thomas Heyman, Riccardo Scandariato, Fabio Massacci and Wouter Joosen, "Towards a quantitative assessment of security in software architectures", Proc. 13th Nordic Workshop on Secure IT Systems, 2008.

[14] Y. Sun and A. Kumar, "Quality-of-Protection (QoP): a quantitative methodology to grade security services", Proc. 28th Int'l Conf. on Distributed Computing Systems Workshops, 2008, pp. 394-399.

[15] N. Artaiam and T. Senivongse, "Enhancing service-side QoS monitoring for Web services", Proc. Ninth ACIS Int'l Conf. on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008, pp. 765-770.

[16] CVSS: Common Vulnerability Scoring System version 2.0. Available at http://www.first.org/cvss/cvss-guide.html

[17] San-Yih Hwang, Haojun Wang, Jian Tang and Jaideep Srivastava, "A probabilistic approach to modeling and estimating the QoS of Web-services-based workflows", Information Sciences, 177, 23, Dec 2007, pp. 5484-5503.

[18] S. A. Butler, "Security attribute evaluation method: a cost-benefit approach", Proc. 24th Int'l Conf. on Software Engineering (ICSE'02), 2002, pp. 232-240.

[19] W. van der Aalst, A. ter Hofstede, B. Kiepuszewski and A. Barros, "Workflow patterns," Distributed and Parallel Databases, 14(3):5–51, July 2003.

[20] N. Russell, A.H.M. ter Hofstede, D. Edmond and W.M.P. van der Aalst, "Workflow data patterns", QUT Technical Report, FIT-TR-2004-01, Queensland University of Technology, Brisbane, 2004.

[21] I. Mura, A. Bondavalli, X. Zang and K. S. Trivedi, "Dependability modelling and evaluation of phased mission systems: a DSPN approach", Proc. IEEE 7th IFIP Int. Conf. on Dependable Computing for Critical Applications, San Jose, CA, USA, 1999, pp. 299–318.

[22] Fabio Massacci and Artsiom Yautsiukhin, "An algorithm for the appraisal of assurance indicators for complex business processes", Proc. 3rd Workshop on Quality of Protection, Oct. 29. 2007, pp.22-27.

[23] Charge it! How to process Online Credit Card Transactions. Available at http://www.aureliodamico.com/documents/students/ProcessingCCTransactions.pdf