Journal of Universal Computer Science, vol. 18, no. 6 (2012), 775-797
submitted: 15/10/11, accepted: 15/2/12, appeared: 28/3/12 © J.UCS

Qos-Security Metrics Based on ITIL and COBIT Standard for Measurement Web Services

Pattama Charuenporn
(Software Systems Engineering Laboratory
Department of Mathematics and Computer science, Faculty of Science
King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand
s1062903@kmitl.ac.th)

Sarun Intakosum
(Software Systems Engineering Laboratory
Department of Mathematics and Computer science, Faculty of Science
King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand
kisarun@kmitl.ac.th)

Abstract: Web Services have been widely adopted in business projects, and almost all Web Service developers agree that security factors are the principal components that must be taken into consideration. A large number of security metrics and measurements is available for specific business needs, and the best practice for different business demands is therefore needed if the quality of service security metrics (Qos-SM) is to be developed. This research proposes a new way of developing Qos-SM using Qos ontology mapping with two information system standards, COBIT and ITIL, as a result of which new Qos-SM are developed. In order to prove the correctness and precision of the metrics, the researchers have used the metrics to measure the level of security quality from Web service data sets. The experimental results, based on vector analysis, show that the same level of security quality is attained with both of the metrics developed and the metrics from previous research. This research also represents the metrics in the form of a class diagram, thus facilitating its application in the organization.

Keywords: Quality of service, ITIL, COBIT, Security Metrics and Measurement, Security for web service
Category: D.2.4, D.2.8, D.2.m

1 Introduction

Web Services technology based on the concept of Service-Oriented Architecture (SOA) Governance has been widely adopted in most of today’s business organizations, but many organizations have not been successful in its usage. An article in ebizQ Magazine by Michael Stamback [Stamback, 08] stated that the adoption of SOA Governance ended in failure in 49% of the organizations that had attempted to use it. The reason behind this failure is a lack of understanding of SOA operation. Joe McKendrick [McKendrick, 06] stated that problems with the SOA technology begin with the SOA security design. Other problems originate from the network’s stability, which is the principal structure of SOA. Network troubles will inevitably affect the entire SOA. Many business organizations have therefore adopted the quality of service security metric (Qos-SM) to solve SOA’s security failure problems. Attention to Qos-SM attributes allows system developer to understand the principles of Web Service operation. Since there are many Qos-SMs, it is difficult for organizations to select an appropriate security metric. In some cases, the organizations cannot manage their information system in order to calculate the worthiness of investment to satisfy their business needs. Apart from better understanding in applying the security metrics attributes to Web Services, another important factor that can enhance Web Services security for organizations is the best practice for Web Services development which is consistent with organizations' business processes. This research chose COBIT (Control Objectives for Information and related Technology) and ITIL (Information Technology Infrastructure Library) Standards, which are both concepts and guidelines for controlling organizations' technologies. Organizations can use them as the best practice for organizational development. The COBIT Standard consists of four major domains; 1) Plan and Organize, 2) Acquire and Implement, 3) Deliver and Support and 4) Monitor and Evaluate. The system security belongs to the Delivery and Support Domain. The ITIL Standard consists of five processes; 1) Service Strategy, 2) Service Design, 3) Service Transition, 4) Service Operation and 5) Continual Service Improvement. The system security belongs to the Service Design process. Both standards are adopted by a large number of business organizations. In 2008, the relationships between the COBIT and ITIL standards were paired in order to analysis the similarity of the processes. As a result, business organizations can apply these processes to their organizational development. From these relationships, the researcher mapped the relationship between COBIT and ITIL standards to derive semantic and ontology relationships with which to develop the Qos-SM. In order to correctly develop security metrics that are based on COBIT/ITIL standards, the researcher conducted an experiment to identify the value of the Qos-SM by comparing it with two groups of sample from related research. Vector analysis was used to calculate the proximity values between three groups (one from Qos-SM and two from related research). The proximity values of three security metrics will be used to identify security metrics in which a sample group has the highest proximity value based on the COBIT/ITIL standards. The security metrics identified will be adopted as an acceptable security standard to reduce problems that cause the damage to the system. They can also be used for future reference and operation. There are six sections in this paper; 1) Introduction, the state of the problems and glossary, 2) Details of related works: COBIT and ITIL standards, ontology relationships, the mapping of both standards, details of the two sample groups selected for the experiment, and the analysis of additional problems that appeared in related works, 3) Charts of the overall development of quality of service security metrics based on COBIT/ITIL standards, an explanation of quality model matching, a definition of the vector used to derive proximity values in the experiment, an explanation of the experiments carried out by the researcher, a mapping table of the experiment values and the class diagram of the quality of service security metrics, 4) Experimental results, 5) Conclusion and 6) Future works.

1.1 Glossary

- COBIT: The Control Objectives for Information and related Technology provides good practices across a domain and process framework and presents activities in a manageable and logical structure. COBIT's good practices represent the consensus of experts. They provide a set of 34 high-level control objectives, one for each of the IT processes, grouped into four domains: plan and organise, acquire and implement, deliver and support, and monitor and evaluate. Each high-level control objective is subdivided into a list of detailed control objectives. In total, COBIT contains 318 detailed control objectives. [IT Governance, 06]

- ITIL: The Information Technology Infrastructure Library is based on defining best practice processes for IT service management and support, rather than on defining a broad-based control framework. It focuses on the method and defines a more comprehensive set of processes. Additional material in ITIL V3 provides a business and strategic context for IT decision making and for the first time describes continual service improvement as an all-encompassing activity, driving the maintenance of value delivery to customers. [ITGI, 08]
- Business strategy: a set of generic business and IT goals provides a business-related and more refined basis for establishing business requirements and developing the metrics that allow measurement against these goals. [IT Governance, 06]
- Miscellaneous requirements: All requirements associated with parts of the organization {e.g. IT infrastructure, business strategy, enterprise architecture IT and available policies}.
- Qos-security requirements: All requirements related to the Qos-security metric for the measurement of Web services.
- Qos-security ontology: Security metrics that present a relation between the attribute in the ontology method.
- Qos-security metrics: Security metric composed of 5 attributes: Consist of Non-repudiation, Authorization, Auditability, Data-encryption, Authentication
- Qos-security specification: (offer or demand) of a Web service materializes as a set of constraints on a certain set of QoS-security metrics.
- QoS attribute: Measured by one or more QoS metrics, which specify the measurement method, schedule, unit, value range, authority and other measurement details. [Kyriakos, 08]
- IT best practices guideline: In this paper, this is composed of 2 guidelines : ITIL and COBIT

2 Related Works

In this section we describe relevant works related to the implementation approach for Web Services security. In order to discuss and prepare reasons for exploring a comparison of all the procedures investigated, we have chosen the ontology approach and functional quality attributes to create security metrics which can explain respectively: 1) the presentational relationship between COBIT and ITIL for using in organizations; 2) the mapping relationship between ITIL COBIT and Qos-security ontology; and 3) the use of the functional quality attributes approach to retrieve Qos attributes that relate to IT practice guidelines.

2.1 Presentational relationship between COBIT and ITIL for use in an organization

The proposed system consists of two important frameworks, ITIL and COBIT, which were studied and then used to shape the functions of the proposed system. The application of IT has become more and more important to the strategy and business processes of many entities. In recent years it has therefore become increasingly evident that there is a need for a reference framework to ensure that the enterprise's information and related technology support for its business objectives and its resources are used responsibly, and that its risks are managed appropriately [IT Governance, 00]. Successful organizations therefore require a basic understanding of the risks and constraints of IT at all levels within the enterprise in order to achieve an effective direction and adequate controls. COBIT is an accepted standard for IT security and control practices that provides a reference framework for management, users and IS audit, control and security practitioners. The COBIT framework encompasses seven information requirements and four IT resources. It provides a set of thirty four high-level control objectives, one for each of the IT processes, grouped into four domains: plan and organise, acquire and implement, deliver and support, and monitor and evaluate. ITIL has four topics that are related to ITSM (IT service management). It describes 1)IT service management functions, 2)activities and organizational structures, 3)strategic and sourcing concerns, 4) integration with the business. Kim [Kim, 03] described the ITIL framework as a 'process-based approach to IT activity' and stated that ITIL is not focused on technology but is rather based on processes critical to organizations. The ITIL framework defines a set of best practices for these processes. Kim [Kim, 03] suggested that organizations use ITIL to identify and improve business processes, using a set of best practices and then matures these processes by using appropriate technologies. The relationship between COBIT and ITIL is therefore shown in figure 1 below. COBIT is generally used to determine the tactical level of the organization, while ITIL prepares data in order to implement the operational strategy; these two level frameworks prepare data for a balance scorecard to implement a strategic level.

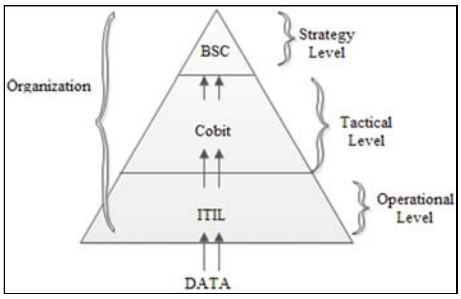
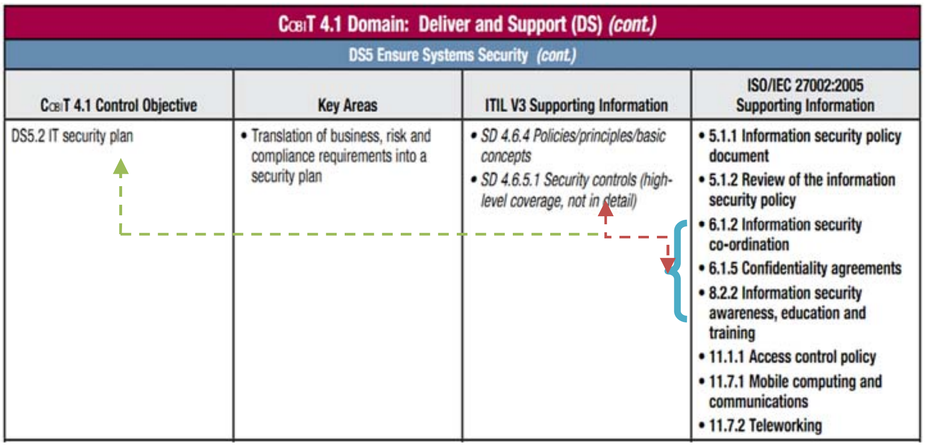


Figure 1: Relationship between COBIT ITIL and the Balance scorecard [Da Cruz and Labuschagne, 06]

Figure 1 shows a detailed relationship description of ITIL and COBIT. This relationship demonstrates that our model can be used in this relationship to create Qos-SM in order to achieve specific business goals in an organization. ITGI [ITGI, 08] shows the reasons for mapping COBIT and ITIL. This paper states that COBIT and ISO/IEC 27002 help to define what should be done and ITIL provides the know-how for service management aspects, and then identifies the process areas in IT that are critical for delivering value and managing these risk areas. The COBIT process framework can be used as a basis and be underpinned by ITIL's definition of key service delivery processes and ISO/IEC 27002's security objectives. This paper gives reasons for mapping each of the COBIT's thirty four IT processes and control objectives that have been mapped to specific sections of ITIL; whereas reverse mapping shows how ITIL V3 key topics map onto COBIT 4.1.We then use the above reasons stated in ITGI to create an ontology and a mapping relationship between COBIT, ITIL and Qos ontology, and use the supporting information in Table 1 to map the COBIT and ITIL ontologies which are described in [section 2.2].



CobiT 4.1 Domain: Deliver and Support (DS) *(cont.)*

DS5 Ensure Systems Security *(cont.)*

CobiT 4.1 Control Objective	Key Areas	ITIL V3 Supporting Information	ISO/IEC 27002:2005 Supporting Information
DS5.2 IT security plan	• Translation of business, risk and compliance requirements into a security plan	• *SD 4.6.4 Policies/principles/basic concepts* • *SD 4.6.5.1 Security controls (high-level coverage, not in detail)*	• 5.1.1 Information security policy document • 5.1.2 Review of the information security policy • 6.1.2 Information security co-ordination • 6.1.5 Confidentiality agreements • 8.2.2 Information security awareness, education and training • 11.1.1 Access control policy • 11.7.1 Mobile computing and communications • 11.7.2 Teleworking

Table 1: Mapping between ITIL, ISO/IEC 27002 and COBIT[ITGI, 08]

2.2 Mapping the relationship between ITIL COBIT and Qos-security ontology

The Qos ontology has recently used many ontological approaches such as DAML-Qos ontology, on Qos ontology, Qos-MO ontology, WSMO-Qos ontology [Tran, 08]. This means that developers need to know the details of most ontologies. It is thus difficult to choose a suitable Qos ontology for each business process problem. Many research works propose an automatic process with which to create Qos for the matching of business processes. The above ideas are applied in combination with the ontological approach to explain a method with which to map relationships. Figure 2-5 demonstrates the mapping relationship between the COBIT ontology, the ITIL ontology and the Qos ontology. In [Charuenporn, 10] the relationships between the ITIL ontology and the COBIT ontology are shown in Fig. 2-3.

2.2.1 ITIL ontology

We selected two processes from the ITIL standard (service delivery and service support) and applied the ontology approach to them for further analysis.

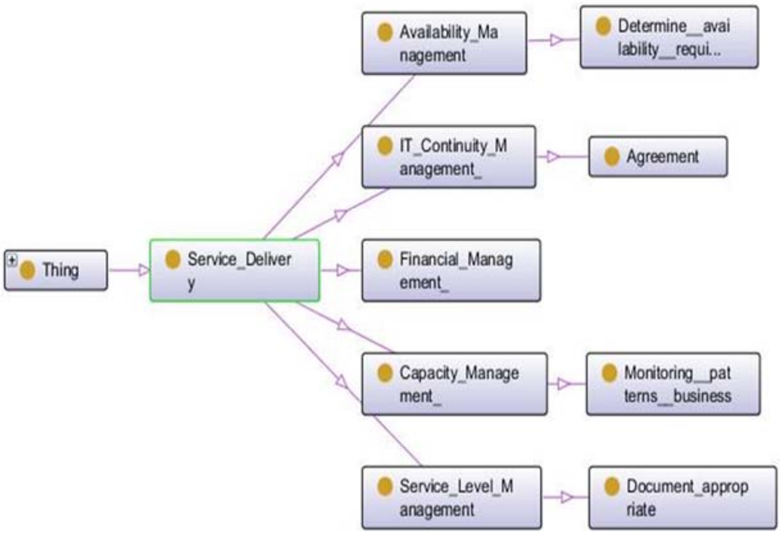


Figure 2: Service Delivery ontology [Charuenporn, 10]

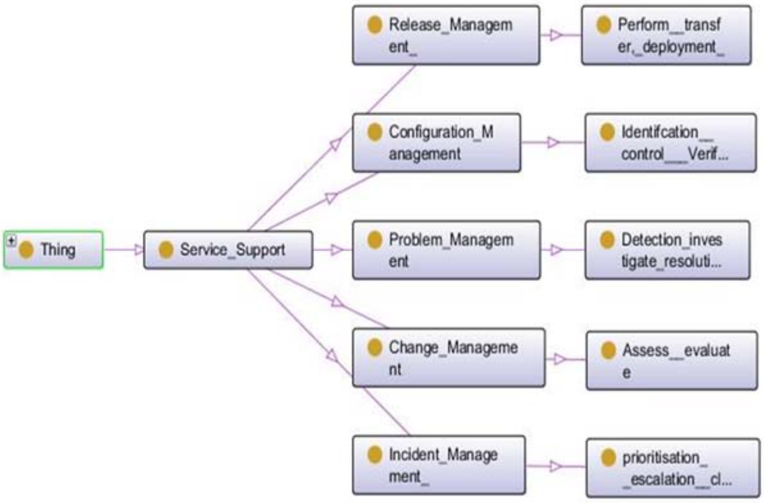

Thing
Service_Support
Release_Managem ent_
Perform__transf er,_deployment_
Configuration_M anagement
Identifcation__ control__Verif...
Problem_Managem ent
Detection_inves tigate_resoluti...
Change_Manageme nt
Assess__evaluat e
Incident_Manage ment_
prioritisation_ _escalation__cl...

Figure 3: Service Support ontology[Charuenporn, 10]

Service delivery consists of six components: 1) service level management, 2) financial management, 3) capacity management, 4) IT service continuity management, 5) availability management and 6) security management. Service support consists of five components: 1) incident management, 2) problem management, 3) configuration management, 4) change management and 5) release management.

2.2.2 COBIT ontology

We selected one domain from the COBIT standard (service delivery and support) and applied the ontology approach to it for further analysis.

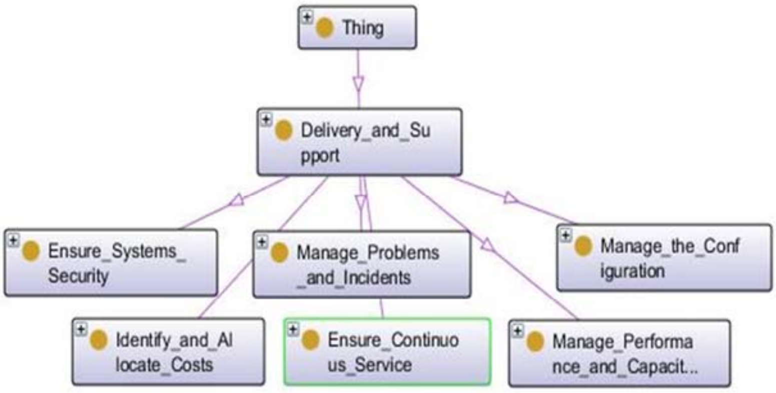

Thing
Delivery_and_Support
Ensure_Systems_Security
Manage_Problems_and_Incidents
Manage_the_Configuration
Identify_and_Allocate_Costs
Ensure_Continuous_Service
Manage_Performance_and_Capacit...

Figure 4: Delivery and Support [Charuenporn, 10]

Service delivery and support consists of six components: 1) ensure system security, 2) manage problems and incidents, 3) manage the configuration, 4) identify and allocate cost , 5)ensure continuous service and 6) manage performance and capacity.

[Charuenporn, 10] stated that the relationship in Figure 4 is suitable for use in specific relationship patterns, and it is easy to identify that some attributes of the COBIT ontology and the ITIL ontology relate to more than one attribute of the QOS ontology, as is described in Figure 5.

Finally, this relationhip has been specifically designed for Qos requirements mapping with the ITIL and COBIT ontologies, so all of the concerns relating to both Qos requirements and IT practices have been taken into consideration. However, this study will be concerned with some approaches that relate to the mapping of other quality attributes. Particularly in the case of security metrics, we shall consider the functional quality attributes (FQAs) approach to choose specific security metrics, along with the Seung [Seung, 08] approach of Qos for Web Services which will be presented to deal with the specification and management of Qos-SM for Web Services.

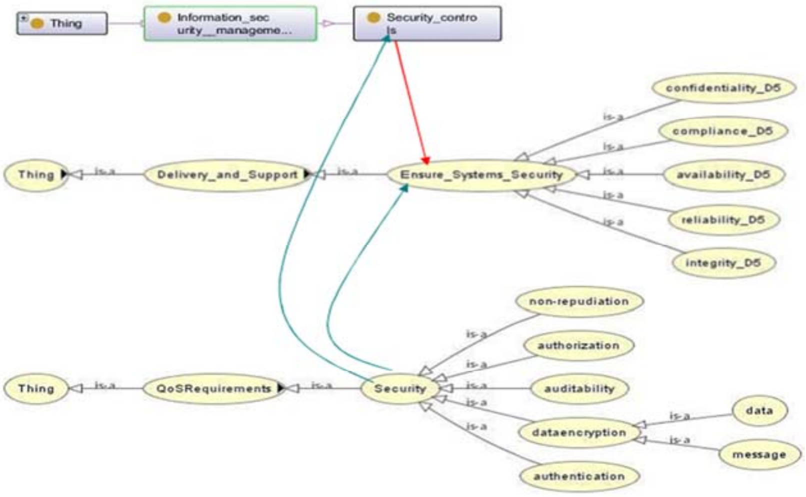


Figure 5: Approach of mapping Qos attribute [Charuenporn, 10]

2.3 Concern and Dependencies Quality Attribute Graph

This section presents reasons for creating concern and dependencies between quality attributes (security attributes and others). In Pinto‘s research [Pinto, 11], the functional quality attributes (FQAs) approach is to justify the significance of specifying parameterizable architectural patterns. Figure 6 illustrates several sets of related concerns for each attribute and the interactions between different security attributes. It has two difference circle notations: the circles in dark grey are examples of quality attributes and the circles in white concern the categories of each attribute. The dashed lines represent dependencies and/or interactions between attributes. [Juristo et al., 03] and [Barbacci et. al., 95]. In Figure 6, it will be noted that most concerns, for example the confidentiality concern (Security FQA), are related to security attributes and are required to achieve the contextual help concern (Usability FQA).

In conclusion, there must be a clear idea of each of the concerns responsible for several FQAs and those that are exclusive to one particular FQA. It is also important that the decomposition of concerns for each FQA should be independent of the application, and it will therefore be possible to define a repository of reusable solutions (reusable architectural patterns) for each FQA. Moreover, the relationships between the different FQAs do not normally depend on the final application, so they can also be modelled as pre-fabricated solutions (reusable architectural patterns). However, there are some situations in which these architectural patterns need to be parameterized [Pinto, 11]. We will assume that FQA is suitable for each business model. Nevertheless, we can apply security metrics, particularly confidentiality, availability and reliability, to mapping with security relationships under ITIL and COBIT.

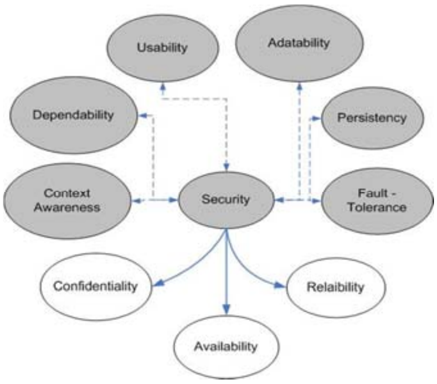


Figure 6: Quality attribute graph [Pinto, 11]

3 Approach for calculated relation metrics

This section explains that the Ontological approach is, overall, flexible in creating a relationship between IT practice guidelines and Qos characteristics. The detailed explanation of how to calculate the Qos-SM method in order to provide a model that is associated with Qos-SM meaning will be discussed. The current approaches that create the security metrics under IT practice guidelines will also be presented. The method will be described first, followed by different definitions of the Qos. The Qos model's appearance will then be explored. At the end of this section, we evaluate the security metrics by testing the data set to prove the concept.

3.1 Approach for creating security metrics under ITIL and COBIT

In this section we show how we applied the security ontology for Qos and why we chose ITIL and COBIT to generate Qos-SM for organizations. Figure 7 shows an overview of the proposed methodology and describes how we used a security ontology and existing reasoning to support ITIL and COBIT based Qos-SM.

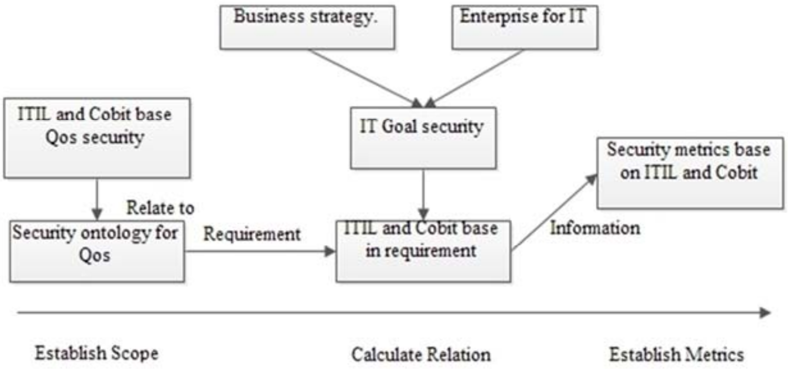


Figure 7: Approach for creating security metrics under ITIL and COBIT

In the first step: (Establish scope of measurement) we use ITIL and COBIT, which demands the accomplishment of specific Qos-security requirements. Since each Qos-security ontology control is related to one or more ITIL and COBIT, we use the relationship with Qos-security ontology controls and their miscellaneous requirements to establish the scope of measurement.

In the second step: (Calculate relation metrics) we use reasoning to extract knowledge concerning the organization's control implementations from the Qos-security ontology based on COBIT and ITIL, and then calculate the weight of miscellaneous items (this step assumes that everything associated with parts of the organization {e.g. IT infrastructure, business strategy, enterprise architecture IT and available policies}) is modeled and mapped onto our ontological model.

In the third step: (Establish quality metrics) the miscellaneous information generated from Step 2 is used together with security information concerning existing control implementations (extracted from the security ontology) to generate the security quality metrics.

3.2 The Quality Model

The current proposed Web Services model is largely unregulated based on UDDI registries. IBM, Microsoft, and SAP have each announced that their current online services will be discontinued because of the promotion of the public UDDI [Nicholson, 06]. This is one example that explains the importance of addressing the quality of service (Qos). The proposal is that this model, in order to present a new model, can replace the current derestricted UDDI registries. This model combines best IT practice guidelines to match with Qos together. The current proposed derestricted registries can offer services to people to whom the quality of service is not important. The derestricted registries based on the model presented here can serve applications that need quality of service assurance. There are five roles in this proposed model: 1) Web Services provider, 2) Web Services consumer, 3) Qos model matching, 4) the new UDDI registry and 5) Qos registry. As before, the Web Services provider offers Web Services by publishing the service in the registry; the Web Services consumer needs the Web Services offered by the provider; the new UDDI registry is a repository of registered Web Services with look-up facilities; the new certifier's role is to verify the service provider's Qos claims, as described in Figure 8. Qos model matching retrieves the IT policy from database and then compares it with Qos in the new UDDI. If the service provider registry service is in the Qos registry, the Qos registry applies the metrics and then sends the Qos listing to UDDI for registering. The proposed new model differs from the current model in that it contains information about the functional description of both the Web Service and its associated Qos registered in the repository. Look-up can be created with a functional description of the desired Web Services, with the required Qos attributes as look-up constraints. The new role in this model is the Web Services Qos model matching which does not exist in the original UDDI model. The model verifies the claims of Qos for a Web Service before its registration. The details of the Web Service provider, the consumer and invocation are shown in Figure 8.

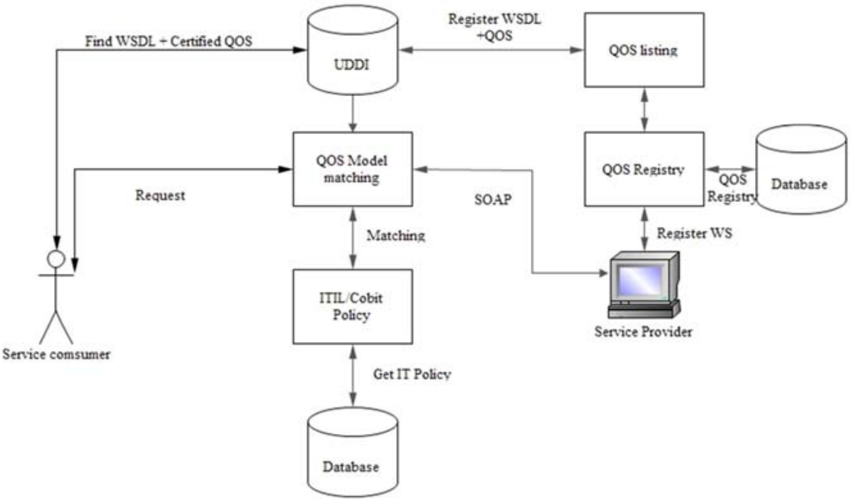


Figure 8: Quality model matching

In this Qos model, the Qos matching algorithm uses a vector space model. The key idea of the Qos matching algorithm is to find the nearest (Qos_i) specifications of the consumer to the ITIL and COBIT policy, where $1 \leq i \leq n$,i = 1, 2, . . . , k is a quality criterion, as explained in Figure 9. Thus, for a set of Web Services (WS) that have the same functional properties, where WS = $\{ws_1, ws_2, ..., ws_n\}$, n is the ordering of the Web Services. Each Qos constraint consists of certain quality criteria: the non-functional, functional and quality properties, and these definitions extend from the vector space method to incorporate the Qos model as described in Figure 9:

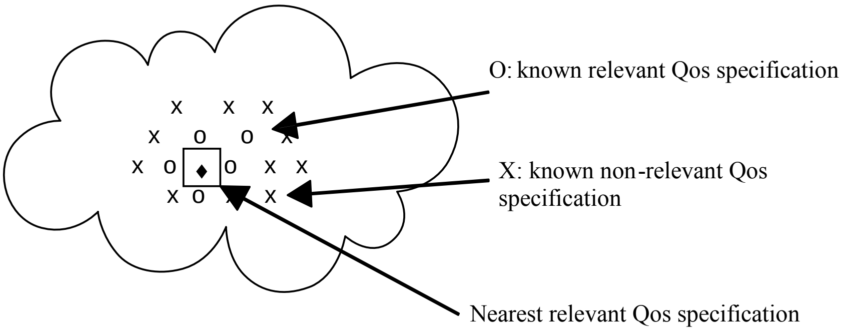

O: known relevant Qos specification
X: known non-relevant Qos specification
Nearest relevant Qos specification

Figure 9: Vector space method

In the Qos model, we can represent this concept in two definitions as follows: 1) the service consumer and 2) Qos model matching.

3.2.1 Service consumer

A service consumer sends a service request message to a service provider and the service provider returns a response message to the service consumer. The definition is shown as follows:

Definition 1: A service consumer C consists of a set of concepts used to describe the Web service in which a group of services is inserted.A set of service consumers can be written as C = (NF, F,Q), where NF is a non-functional property, F represents a functional property and Q is a quality property.

3.2.2 Qos model matching

This model combines best IT practice guidelines to match Qos together. The definition is shown as follows:

Definition 2: For two different services that have: 1) Qos based on related Qos requirements and 2) Qos based on ITIL and COBIT. The advertisement of a service can be depicted as follows:

- A service consumer based on related Qos requirements is denoted as

$$C_{ri} = (NF_{ri},F_{ri},Q_{ri}) \qquad (1)$$

Based on related Qos requirements, C_r denotes a service consumer, NF_r denotes a non-functional property, F_r represents a functional property, Q_r is a quality property and i denotes a Web service that is associated with a service consumer C_r from the sample data. In this paper, we chose two samples of Qos-SM data as follows:

Example one [Pinto, 11]: $C_{r1} = (NF_{r1},F_{r1},Q_{r1})$

Example two [Seung, 08]: $C_{r1} = (NF_{r2},F_{r2},Q_{r2})$

Where C_{r1} and C_{r2} are the service consumers from the aforementioned research [Pinto, 11] and [Seung, 08].

- A service consumer based on ITIL and COBIT is denoted as

$$C_{ct} = (NF_{ct},F_{ct},Q_{ct}) \qquad (2)$$

Based on ITIL and COBIT standards, C_{ct} denotes a service consumer, NF_{ct} denotes a non-functional property, F_{ct} represents a functional property and Q_{ct} is a quality property.

We plotted the value of service consumers (C_{r1}, C_{r2} and C_{ct}) on the axis as depicted in Figure 10.

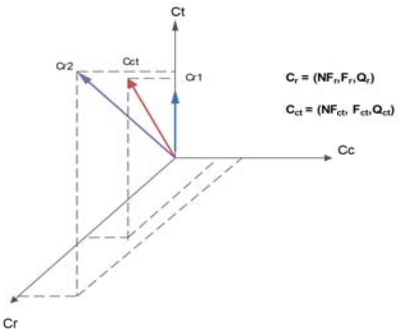


Figure 10: Vector model of service consumers (C_{r1}, C_{r2} and C_{ct})

We then analyzed the relationship between C_{r1}, C_{r2} and C_{ct} by calculating and comparing the proximity value. The proximity value is a cosine value between C_{r1}, C_{r2} and C_{ct} as shown in equation 3.

$$\cos\theta = \frac{\overrightarrow{C_{r\imath}}\cdot\overrightarrow{C_{ct}}}{C_{ri}\cdot C_{ct}} \tag{3}$$

Where $\overrightarrow{C_{r\imath}} \cdot \overrightarrow{C_{ct}} = (C_{rix}C_{ctx} + C_{riy}C_{cty} + C_{riz}C_{ctz}) \cos\theta,$ (4)

And $$C_{ri} = \sqrt{C_{rix}^2 + C_{riy}^2 + C_{riz}^2} \tag{5}$$

$$C_{ct} = \sqrt{C_{ctx}^2 + C_{cty}^2 + C_{ctz}^2} \tag{6}$$

Where x, y and z denote the dot product value on each axis.

If $\cos\theta \approx 1$, this indicates that Qos-SM is exactly the same as ITIL/COBIT.
If $\cos\theta \approx 0$, this indicates that Qos-SM is completely different to ITIL/COBIT.

3.3 Calculate Proximity Value

As a proof of concept of Quality models, we selected two factors, which are confidentiality and availability, from three samples ([Pinto, 11], [Seung, 08] and ITIL/COBIT standards) as follows:

- Confidentiality (CC): This ensures that critical and confidential information is withheld from those who should not have access to it, for example, permitting access to critical and sensitive data only to authorized users, Number and type of malicious code prevented.

We selected three quality attributes which are authentication, authorization and encryption for testing (a, au and e). The acronyms for these quality attributes for CC_{r1}, CC_{r2} and CC_{ct} have been defined as (CC_{r1}a, CC_{r1}au, CC_{r1}e), (CC_{r2}a, CC_{r2}au, CC_{r2}e) and (CC_{ct}a, CC_{ct}au, CC_{ct}e).

- Availability (CA): This ensures that IT services and infrastructure can resist and recover from failures caused by error, deliberate attack or disaster, for example, a number of systems in which security requirements are not met, Time to grant, change and remove access privileges.

We selected three quality attributes which are: successful execution rate, duration time and integrity for testing (a, e and t). The acronyms for these quality attributes for CA_{r1}, CA_{r2} and CA_{ct} have been defined as (CA_{r1}a, CA_{r1}e, CA_{r1}t), (CA_{r2}a, CA_{r2}e, CA_{r2}t) and (CA_{ct}a, CA_{ct}e, CA_{ct}t).

These relationships are summarized in Table 2: The symbols are all the same but different in meaning.

	Confidentiality (CC)	**Availability (CA)**
[Pinto ,11])	(CC_{r1}a, CC_{r1}au, CC_{r1}e)	(CA_{r1}a, CA_{r1}e, CA_{r1}t)
[Seung ,08]	(CC_{r2}a, CC_{r2}au, CC_{r2}e)	(CA_{r2}a, CA_{r2}e, CA_{r2}t)
[ITIL and COBIT]	(CC_{ct}a, CC_{ct}au, CC_{ct}e)	(CA_{ct}a, CA_{ct}e, CA_{ct}t)

Table 2: Equation of confidentiality and availability

In the experiments, we used XML script from an Emergency Service Provider [XMethod, 11], and then tested data for each criterion from Table 2. We tested under a variety of test sets in the same environment. This section aims to describe the methods used to validate the approach. The main purpose of these experiments is to analyze the feasibility of the proposed approach with regard to the security metrics. Test data are presented for each quality criteria. Their corresponding value types are in sequence and are shown in Table 2. In Table 3, we show the results from the execution of the experiments.

	Confidentiality (CC)	**Availability (CA)**
[Pinto ,11]	(3a, 4au, 3e)	(3a,4e,5t)
[Seung ,08]	(2a, 5au, 2e)	(4a,5e,5t)
[ITIL and COBIT]	(4a, 5au, 3e)	(5a,4e,6t)

Table 3: Result of Execution from Emergency Service Provider [XMethod, 11]

Upon calculating the proximity value between all of them, following equations 1, 2, 3 and 4, we can conclude that if cos ɵ ≈ 1, then the Qos-SM is similar to ITIL and COBIT. The results of the experiment show that the meaning of Qos-SM by Pinto is the nearest meaning of Qos-SM based on ITIL and COBIT. It shows that Qos-SM is based on ITIL and COBIT and the description of the metrics reference follows Pinto. Based on this result it would be possible to conclude that our Qos model is dynamic and uses all new metrics which can be fully adapted to the current distributed network environment.

3.4 Create Security description based on ITIL and COBIT

Following the result shown in [section 3.3], we can explain the relationship between ITIL, COBIT and Qos-SM, and then give weights to map the relationship. The value of the weight can be one of the following: 1, 0.5, and 0, which signify fully associated, rarely associated, and not associated. This is described in Table 4.

QOS Attribute	Confidentiality		Availability		Reliability	
	[Pinto, 11]	[Seung, 08]	[Pinto, 11]	[Seung, 08]	[Pinto, 11]	[Seun, 08]
Authentication	1	1	0	0	1	0
Encryption	1	0	0	0	1	0
Non-Reputation	0	0	0	0	1	0
Authorization	1	0	0	0	1	0
Audit	0	0	0	0	1	0
Integrity	0	0	1	1	1	0

Table 4: Weight for finding metrics

According to Table 4, we will use all of the weights equal to 1 to explain each Qos attribute. It can be concluded that confidentiality has a weight equal to 1. In authentication, encryption and authorization, availability has a weight equal to 1. In integrity and reliability it has weight equal to 1. In authentication, encryption, non-reputation, authorization, audit, integrity it has a weight equal to 1. in order to remain consistent with the weight for the experiment shown in [section 3.3], we present a new definition of Qos-SM based on ITIL and COBIT, as is shown in Table 5.

IT guideline Class	Control Objective Class	Security QOS Class	QOS Attribute
ITIL	Information security management	Confidentiality Availability Reliability	Authentication Encryption Non-Reputation Authorization Audit Integrity
COBIT	Ensure system security		

Table 5: Mapping Security Qos with Business Characteristic

All above characteristics consist of issues related to the protection and privacy of the quality model containing all the metrics. In Table 5, the IT guideline consists of two classes: ITIL and COBIT. One detail of ITIL particularly associates Qos-SM in the area of service design; the main reference of service design is information security management. It consists of information security policy, Information Security Management System (ISMS) and key performance indicators. According to COBIT, the main reference associated with security is to ensure system security. It consists of two primary (confidentiality and integrity) and three secondary information criteria (availability, compliance and reliability); we used all of ITIL and COBIT's components in order to create Qos-SM.

3.5 Represent Qos-security metrics in Class diagram

The model for Qos-SM is represented by the UML diagram. Because UML is a tool for defining the structure of a system, it is a very useful way in which to manage large, complex systems, since it has a clearly visible structure that makes it easy to introduce new people to an existing project. The UML was launched in 1995 and adopted as an industrial standard by the OMG in 1997 with UML 1.1 version [OMG, 06]. UML 2.0 was launched in 2003 [OMG, 03]. This UML version is used to describe the Qos and its concepts in this section. In Figure 11, the overview of the business model and Qos model are represented in the class diagram. In the business model , the business goal defines the business requirements which are sets of generic businesses and IT goals. A business model may contain one or more business class(es) and each business class consists of its own business object(s). In order to develop the Qos Model, it is necessary to define the Qos goal which support the business goal. The Qos Class and attributes that relate to the Qos goal will be selected. The Qos model may relate to one or more business model, each Qos model may contain one or more Qos class(es) and each Qos class consists of its own Qos object(s)/attributes.

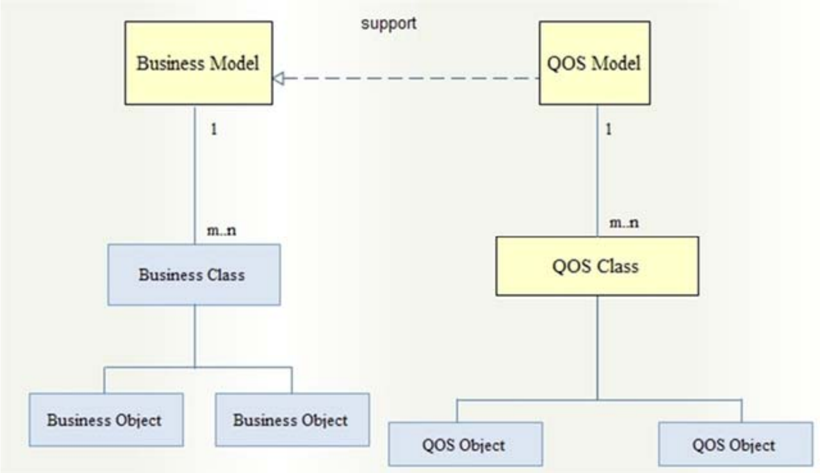


Figure 11: Qos Model - Class diagram

As previously mentioned, we chose security metrics to experiment and perform security metrics in the class diagram, as is depicted in figure 12.

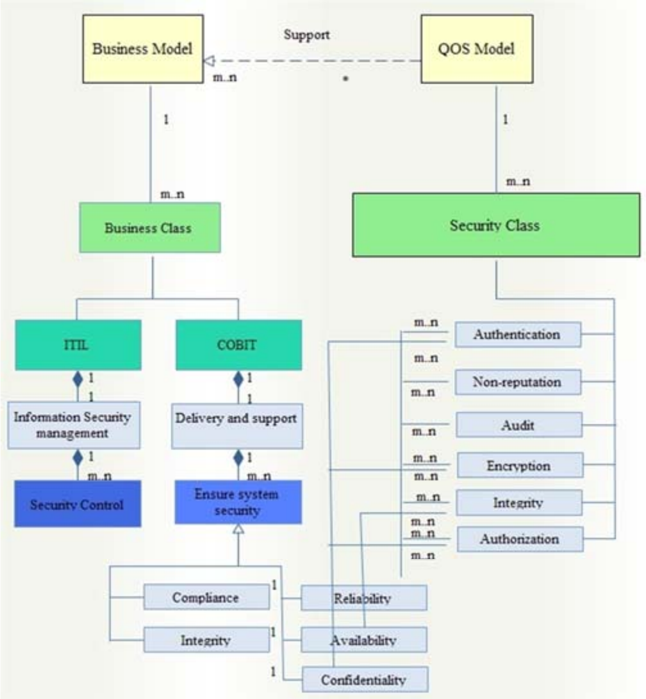
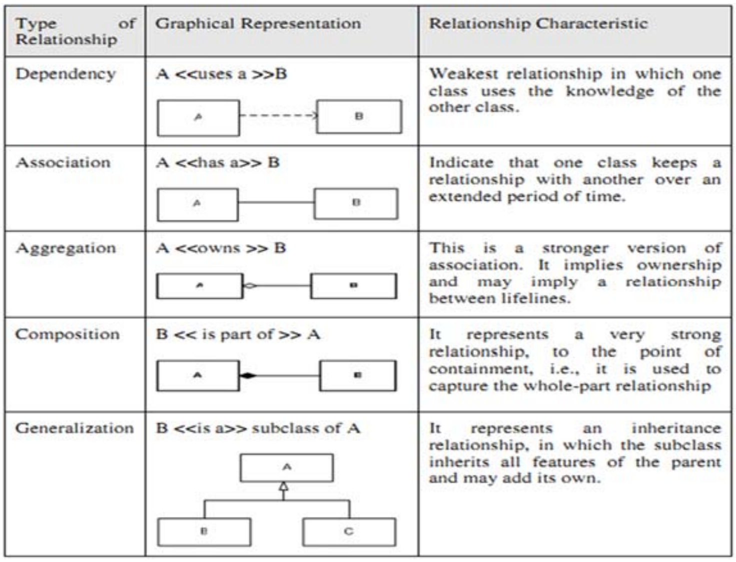


Figure 12: Security Class diagram

Figure 12 illustrates the relationship between business and Qos models. ITIL and COBIT were selected as business classes to support businesss goal. Information security management is a part of ITIL and security control is a part of Information security management. According to COBIT, we can expand the object to ensure the system‘s security. It can inherit five information criteria that comprise the COBIT guideline. For Qos model, we selected security as a Qos class, and certain security attributes such as authentication, authorization and encryption as Qos objects. Follwing the relationship between business and security classes in the FQAs method [Pinto, 11], we can conclude that there are three objects in the business class that relate to six objects in the security class. We used these three objects (reliability, availability and confidentiality) to measure Web Services and draw conclusions about security metrics based on ITIL and COBIT.

In Figure 11-12, the classes are related to each other through relationships. Each UML relationship represents a different type of connection between the classes [Pitman, 05]. There are many different types of relationships. Those most frequently used in meta-modeling language have been listed below. The most common relationships and their characteristics are provided in Figure 13.



Type of Relationship	Graphical Representation	Relationship Characteristic
Dependency	A <<uses a >>B	Weakest relationship in which one class uses the knowledge of the other class.
Association	A <<has a>> B	Indicate that one class keeps a relationship with another over an extended period of time.
Aggregation	A <<owns >> B	This is a stronger version of association. It implies ownership and may imply a relationship between lifelines.
Composition	B << is part of >> A	It represents a very strong relationship, to the point of containment, i.e., it is used to capture the whole-part relationship
Generalization	B <<is a>> subclass of A	It represents an inheritance relationship, in which the subclass inherits all features of the parent and may add its own.

Figure 13: The most common UML relationships and characteristics [Selic and Schmuller, 04], [Pilone, 05]

4 Experiment Result

In accordance with the results associated with the experiment in [section 3], we can define the Qos specification based on ITIL and COBIT standards. The specification will focus on defining IT security policies, plans and procedures, monitoring, detecting, reporting and resolving security vulnerabilities and incidents. It can be used as a reference for another type of Qos model. In order to apply the specification, five components must be defined as described in Table 6:

<Name>	Name of the criterion, resolving security vulnerabilities and incidents [IT Governance ,00]
<Description>	1. Understanding security requirements, vulnerabilities and threats 2. Managing user identities and authorisations in a standardised manner 3. Testing security regularly
<Rationale>	Managing systems security includes both physical and logical controls that prevent unauthorized access. These controls typically support authorization, authentication, non-repudiation, data classification and security monitoring. Deficiencies in this area can have a significant impact on financial reporting. For instance, insufficient controls over transaction authorization may result in inaccurate financial reporting [IT Governance ,06].
<Metrics>	Addresses the scale of measurement.
<How to measure>	1. Number of incidents damaging the organization's reputation with the public. 2. Number of systems in which security requirements are not met 3. Number of violations in segregation of duties

Table 6: Reference for another type of Qos model

We also found the quality attributes that need to be considered for each selected security class (from Qos model). We then developed the metrics for measuring the quality of Web Services based on IT practice guidelines. The relationships between security classes and quality attributes are depicted in Figure 14.

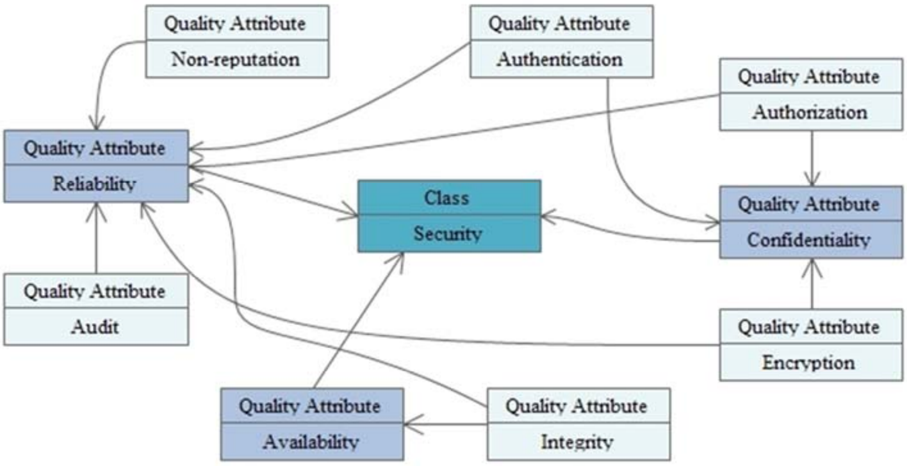


Figure 14: Relationships between security classes and quality attributes

Figure 14 illustrates that quality represents a measurable non-functional aspect of a service within a given domain. It indicates that the quality concept relates to multiple domains. For Web Service security, the security metric with the following quality attributes will be concerned; confidentiality and availability.We can also apply the concept of quality specification to these to quality attributes, as is shown in Tables 7 and 8.

Confidentiality	Understanding and support authorization, authentication and Encryption.
Rationale	Data should be transmitted through Web service protocols.
Metrics	Presentation to percentage.
	Composed of: 1.Authentication: Users (or other services) who can access service and data should be authenticated. 2.Authorization: Users (or other services) should be authorized so that only they can access the protected services. 3.Data encryption: Data should be encrypted.
How to measure	Test quality of Web Services under three factors : Authentication, Authorization and data encryption.

Table 7: Description of confidentiality

Availability	Ensure that critical and confidential information is withheld from those who should not have access to it.
Rationale	-
Metrics	Presentation to percentage.
	Composed of: 1.Authentication: Authentication that can be used with Web services ranging from username/passphrases to client and server side certificates. 2.Authorization: means that only an authenticated entity of a WS can access the information resources needed in order to use the service.
How to measure	Test quality of Web services under a quality: Integrity.

Table 8: Description of availability

5 Conclusions

Qos-SM is the cornerstone of the proposed Web Service Security roadmap and deals with specific security considerations for public Web Services. Thus far, the Qos-SM is an isolated specification but it will be complemented with several other specifications (especially according to business process) in the future. In this paper, we present a new approach for creating Qos-SM for the measurement of Web Services based on IT practice guidelines (ITIL and COBIT) and show the relationship between Qos ontology mapping with the ITIL and COBIT ontologies, and then verify the consistency of security policies in the Qos ontology and the ITIL,COBIT policies by abstracting them. The possible abstraction method must be defined by the testing method. Defining an abstraction method of the relationship between Qos and ITIL,COBIT is constrained based on the semantics of the language. In part of the testing method, we created Qos-SM as follows. First, we specified a Qos-security ontology and its vocabulary in order to augment the Qos information and present a Qos model with which to create metrics.The Qos model is represented as a new method and creates a new relationship by mapping ontology relations between Qos and ITIL and COBIT into the new model. Second, the validation method has been carried out by using vector analysis to verify the Qos model and testing the correctness of Qos-SM based on ITIL and COBIT. The two samples of security metrics have been chosen for the experiment with Qos-SM. Furthermore, we designed and presented Qos-SM in a class diagram to facilitate its application in organizations.

6 Future Work

In this research, we have proposed a model to generate Qos-security based on two best IT practices, and have shown how the Qos-security ontology can be used to generate concrete and specific organizational knowledge that complies with existing control implementations. Further research will address the identified limitations and will create another Qos metric. We plan to align our Qos ontology based on IT security metric generation with COBIT and ITIL. Furthermore, we will take the evaluation of our concepts from an experiment to a real-world level by applying our concepts in real-world audit scenarios. The planned research activities will constitute our second step towards increasing the degree of automation in the field of IT-security metrics.

Acknowledgement

This work was supported by the Software Systems Engineering Laboratory, Department of Mathematics and Computer science, Faculty of Science, King Mongkut's Institute of Technology, Ladkrabang, Bangkok, Thailand, whom we thank for funding this work.

References

[Barbacci et. al., 95] Barbacci, M. and et. al.: “Quality attributes”; Technical Report, Software Engineering institute,Carnegie Mellon University,(1995) ,also appeared as electronic version, in publications/ December-95-online.

[Braden, 05] “The ITIL foundation exam study guide”;(2005)
www.trainning.com.br/download/ITIL%20Foundations%20Ingles.pdf

[Carimo, 06] Carimo, R.: “Evaluation of UML Profile for Quality of Service from the User Perspective”; Master Thesis Software Engineering Thesis no: MSE-2007-03, August 2006

[Charuenporn, 10] Charuenporn, P., Intakosum, S. : “Combine ITIL and COBIT with QOS ontology to measure Web Services”: Proc of the International Conference on e-Commerce, e-Administration, e-Society, e-Education, and e-Technology (2011), 3136-3147, indicate the page numbering.

[Da Cruz and Labuschagne, 06] Da Cruz, E., Labuschagne, L.: “A new framework for bridging the gap between IT Service Management and IT Governance from a security perspective” ;(2006), Academy of Information Technology at the University of Johannesburg.

[ITGI, 08] ITGI, OGC.: “Aligning Co b iT® 4.1, ITIL® V3 and ISO/IEC 27002 for Business Benefit.”;2008, http://webstore.ansi.org

[IT Governance, 00] COBIT Steering Committee and the IT Governance Institute.: “COBIT®3rd Edition Management Guidelines”; (2000) also appeared as electronic version, in publications IT Governance Instituted,2000,online.

[IT Governance, 06] IT Control Objectives for Sarbanes-Oxley.: “The Role of IT in the Design and Implementation of Internal Control Over Financial Reporting”; 2nd Edition Printed in the United States of America,(2006)

[Juristo et al., 03] Juristo, N., Lopez, M., Moreno, A. M., and Sanchez, M.-I.: “Improving software usability through architectural patterns.”; In ICSE Workshop on SE-HCI, (2003),12–19, indicate the page numbering.

[Kim, 03] KIM G., Sarbanes-Oxley, Fraud Prevention, and IMCA.: “A Framework for Effective Controls Assurance”;Computer Fraud & Security,(2003), vol.2003, no.9, 12-16, indicate the page numbering.

[Kyriakos, 08] Kyriakos Kritikos, Dimitris Plexousakis.: “QoS-Based Web Service Description and Discovery”, ERCIM News (2008),72 , indicate the page numbering.

[McKendrick, 06] Joe McKendrick.: “Planning and implementing SOA” Technology Management and Strategy Report, www.butlergroup.com,2006

[Nicholson, 06] Nicholson,K.: “Is This The Decline of Public UDDI Business Registries?”; http://www.theserverside.net/news/thread.tss?thread_id=38136

[OMG, 03] OMG.: “UML 2.0 Infrastructure Specification”; OMG Adopted Specification ptc/03-09-12, (2003),http://www.omg.org/docs/ptc/03-09-15.pdf

[Pilone, 05] Pilone,D., Pilone,N.: “UML 2.0 in a Nutshell.” ; O'Reilly, 234,(2005) , indicate the page numbering.

[Pinto, 11] Pinto, M., Fuentes, L.: “Modeling Quality Attributes with Aspect-Oriented Architectural Templates”; Journal of Universal Computer Science, vol. 17, no. 5 (2011), 639-669, indicate the page numbering.

[Pitman, 05] D. Pilone, N. Pitman.: "UML 2.0 in a Nutshell"; O'Reilly, 234, (2005), indicate the page numbering.

[Schmuller, 04] Schmuller , J.: “Teach Yourself UML in 24 Hours”, (2004),Third Edition, Sams Publishing

[Selic, 04] Selic, B.: “Tutorial: An overview of UML 2.0”; Proc. of the 6th International Conference on Software Engineering,(2004), Edinburgh, United Kingdom, IEEE Computer Society, 741–742, indicate the page numbering

[Stamback, 08] Michael stamback.: “SOA Governance: Survey Says”, SOA Governance@work, (2008), http://blogs.oracle.com/governance/2008/08/

[Seung, 08] Seung,L,Dong, Shing.: “Web service Qos in Multi-domain”; ICACT 2008, (2008), Proc.10th International Conference on Advanced Communication Technology

[Tran, 08)] Tran,V.: “WS-QosOnto: A Qos Ontology for Web Services.”; DOI 10.1109/SOSE.2008.17, IEEE International Symposium on Service-Oriented System Engineering,(2008) 233–238, indicate the page numbering.

[XMethods, 11] “XMethods”; http://www.xmethods.net/ve2/index.po