IEEE Trans on Cloud Computing, Vol 5, #3, pp, 457-471, June-Sept 2017

Quantitative Reasoning about Cloud Security Using Service Level Agreements

Jesus Luna, Ahmed Taha, Ruben Trapero, and Neeraj Suri

Abstract—While the economic and technological advantages of cloud computing are apparent, its overall uptake has been limited, in part, due to the lack of security assurance and transparency on the Cloud Service Provider (CSP). Although, the recent efforts on specification of security using Service Level Agreements, also known as "Security Level Agreements" or secSLAs is a positive development multiple technical and usability issues limit the adoption of Cloud secSLA's in practice. In this paper we develop two evaluation techniques, namely QPT and QHP, for conducting the quantitative assessment and analysis of the secSLA based security level provided by CSPs with respect to a set of Cloud Customer security requirements. These proposed techniques help improve the security requirements specifications by introducing a flexible and simple methodology that allows Customers to identify and represent their specific security needs. Apart from detailing guidance on the standalone and collective use of QPT and QHP, these techniques are validated using two use case scenarios and a prototype, leveraging actual real-world CSP secSLAdata derived from the Cloud Security Alliance's Security, Trust and Assurance Registry.

Index Terms—Cloud security, security metrics, security quantification, security service level agreements

1 Introduction

Cloud computing drives the vast spectrum of both current and emerging applications, products, and services, and is also a key technology enabler for the future Internet. Its direct economic value is unambiguously substantial but taking full advantage of Cloud computing requires considerable acceptance of off-the-shelf services. Consequently, both security assurance and transparency remain as two of the main requirements to enable Customer's trust in cloud service providers (CSPs).

The lack of assurance and transparency, along with the current paucity of techniques to quantify security, often results in Cloud Customers (in particular Small and Medium-sized Enterprises—SMEs) being unable to assess the security of the CSP(s) they are paying for. Despite the advocated economic and performance-related advantages of the Cloud, two issues arise (i) how can a (non-security expert) SME meaningfully assess if the CSP fulfils their security requirements? and (ii) how does a CSP provide security assurance to Customer organizations during the full Cloud service life cycle?

A commonly implemented approach by many public CSPs has relied on the adoption of "security controls frameworks" as a mechanism to provide their prospective Customers a reasonable degree of security assurance and transparency. This is not a surprise as security practitioners have historically relied on mature frameworks such as ISO/IEC 27002 [1] to certify the security of IT services and products. Nowadays, many CSPs are increasingly adopting Cloud-specific security control frameworks such as the Cloud Security Alliance's Cloud Control Matrix (CSA CCM [2]) and National Institute of Standards and Technology (NIST) Special Publication (SP) 800-53 Revision 4 (R4) [3].

Over the implementation of their security control framework, the CSP can only assume the type of data a Customer will generate and use during the operational phase of the Cloud service; therefore, the CSP is not aware of the additional security requirements and tailored security controls deemed necessary to protect the Customer's data. Customers require the mechanisms and tools that enable them to understand and assess what "good-enough security" [4] means in the Cloud. Customers need to become aware of the changes in security assessment that the Cloud brings, in particular with their need to have a transparent view into the Cloud service acquired. This requirement is critical when assessing if, for example, the SME security requirements are being fulfilled by the controls and certifications implemented by the CSP.

Fortunately, different stakeholders in the Cloud community (e.g., the European Network and Information Security Agency -ENISA [5]-, ISO/IEC [6], and the European Commission [7]) have identified that *specifying security parameters in Service-Level Agreements (termed as secSLA in this article)* is useful to establish common semantics to provide and manage security assurance from two perspectives, namely (i) the security level being offered by a CSP, and (ii) the security level requested by a Cloud Customer. At the same time, the state of the practice predominantly focusses on the methodologies to build and represent these Cloud secSLAs

- *A. Taha, R. Trapero, and N. Suri are with the Department of Computer Science, Technische Universität Darmstadt, Darmstadt 64289, Germany. E-mail: {ataha, rtrapero, suri}@deeds.informatik.tu-darmstadt.de.*
- *J. Luna is with the Cloud Security Alliance, Scotland, United Kingdom, and the Department of Computer Science, Technische Universität Darmstadt, Darmstadt 64289, Germany. E-mail: jluna@cloudsecurityalliance.org.*

Manuscript received 12 Mar. 2015; revised 18 June 2015; accepted 31 July 2015. Date of publication 27 Aug. 2015; date of current version 6 Sept. 2017. Recommended for acceptance by K.-K.R. Choo, O. Rana, and M. Rajarajan. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TCC.2015.2469659

(cf., Section 2), but there is a conspicuous gap on the techniques to manage Cloud secSLAs in order to provide security assurance and improve transparency.

1.1 Contributions

Targeting the core aspect of managing Cloud secSLAs through the use of security metrics, this paper builds upon and extends our research on Cloud security quantification namely, [8] (Quantitative Policy Trees (QPT)) and [9] (Quantitative Hierarchical Process or QHP), to provide the following contributions:

1) The elicitation of requirements needed to quantify and aggregate the security levels provided by the different elements of the secSLA, by presenting two use cases to compare and demonstrate the usefulness of the QPT and QHP techniques (both from the Cloud Customer and the CSP perspectives).
2) Through the use of the QPT and QHP quantification methodologies, this paper contributes the techniques to obtain fine and coarse grained information about security levels using Cloud secSLAs.
3) A side-by-side comparison across QPT and QHP is conducted to provide insights related to their individual and collective capabilities.
4) A sensitivity analysis based on Cloud secSLAs is conducted for helping CSPs (a) determine which parameter most affects the overall security level (according to the Customer's requirements), and (b) provide guidance on the security improvements that should be performed by the CSP in order to achieve the requested security level.
5) As a final contribution, we present the prototype of a decision-making dashboard (implemented as a web service) that can be used by (prospective) Cloud Customers to compare different CSPs based on their offered secSLAs, and with respect to a specific set of security requirements.

1.2 Paper Organization

The paper is organized as follows: after a discussion on related work in Section 2, Section 3 highlights the importance and presents the basic terminology related to Cloud secSLAs. Section 4 overviews and extends two security evaluation techniques developed by our previous research on this field, and that will be applied to quantitatively manage Cloud secSLAs. Section 5 presents two use cases demonstrating the usefulness of quantitatively evaluating Cloud secSLAs with real-world data. An empirical validation of our developed methodology appears in Section 6 followed by an overview discussion in Section 7.

2 Related Works

Multiple approaches are emerging to assess the functionality and security of CSPs. In [10], the authors proposed a framework to compare different Cloud providers across performance indicators. In [11], an Analytic Hierarchy Process (AHP) based ranking technique that utilizes performance data to measure various Quality of Service (QoS) attributes and evaluates the relative ranking of CSP's was proposed. In [12], a framework of critical characteristics and measures that enable comparison of Cloud services is also presented. However, these studies focused on assessing performance of Cloud services, but not their security properties.

While some approaches have focused on specifying Cloud security parameters in secSLAs, fewer efforts exist for quantifying these SLA security attributes. In [13] Henning identified security SLAs with applicable types of quantifiable security metrics for non-Cloud systems, where the paper showed three steps to be followed while developing security SLA metrics: policy analysis, architecture analysis and interviews. These metrics were expanded by Irvine and Levin [14], outlining the term "QoSS" for quality of security service. Based on QoSS, Lindskog [15] defined four dimensions that specify a tunable Cloud security service.

Security requirements for non-Cloud services have been addresed by Casola et al. [16], who proposed a methodology to evaluate security SLAs for web services. Chaves et al. [17] explored security in SLAs by proposing a monitoring and controlling architecture for web services. As pointed out by Chaves et al., it is a challenge to define quantifiable security metrics, but they give examples related to password management, frequency of backups and repair/recovery time. In [18] and [19], the authors propose a technique to aggregate security metrics from a web services' secSLAs. Their approach focused on the process of selecting the optimal service composition based on a set of predefined requirements. However, differing from our research, the authors did not propose any techniques to assess Cloud secSLAs or empirically validate the proposed metrics.

In [20] the authors presented a method for managing the secSLA lifecycle in the context of federated Cloud services. However, they did not elaborate the techniques needed to conduct their assessment/management. In [21] the authors propose the notion of evaluating Cloud secSLA's, by introducing a metric to benchmark the security of a CSP based on categories. However, the resulting security categorization is purely qualitative. In [8] Luna et. al presented a methodology to quantitatively benchmark Cloud security with respect to Customer defined requirements (based on control frameworks). Both works are based on the Reference Evaluation Methodology (REM) [22], which allows to compose security levels of different CSPs without being applicable to secSLAs (in contrast to the methodologies presented in Section 4). In [9] the authors presented a framework to compare, benchmark and rank the security level provided by two or more CSPs. The proposed framework allows both basic and expert users to express their security requirements according to their expertise and specific needs. Our research extends the work in [9] by leveraging the specific notions of Cloud secSLAs, that are adopted from current standardisation efforts and real-world case studies.

3 Cloud secSLAs: Value, Usage, Terminology

In order to develop the full context on the value of secSLAs for security quantification, we present the rationale on SLA usage along with the basic SLA terminology needed to present the contributions of this paper (cf., Section 4). The discussion presented in this paper is based on the Cloud secSLA terminology and structure presented on the latest version of the relevant ISO/IEC 19086 standard [23]. At the time of writing this paper that standard was still on a draft

format, although its terminology and general security components were already stable.

3.1 Why are Cloud (Security) SLAs Important?

Contracts and Service Level Agreements (SLAs) are key components defining Cloud services. According to the ETSI Cloud Standards Coordination group [24], SLAs should facilitate Cloud Customers in understanding (i) what is being claimed for the Cloud service, and (ii) relate such claims to their requirements. Where, better assessments and informed user decisions help increase trust and transparency between Cloud Customers and CSPs.

A recent report from the European Commission [25] considers SLAs as the dominant means for CSPs to establish their credibility, attract or retain Cloud Customers since they can be used as a mechanism for service differentiation in the CSP market. This report suggest an standardised SLA specification aiming to achieve the full potential of SLAs, so the Cloud Customers can understand what is being claimed for the Cloud service and relate those claims to their own requirements.

At the SecureCloud2014[1] an online survey to better understand the current usage and needs of European Cloud Customers and CSPs related to SLAs was conducted by CSA. Almost 200 equally balanced Cloud Customer and CSP responders (80 percent from the private sector, 15 percent from the public sector, and 5 percent from other) provided some initial findings on the use of standardized Cloud SLAs. Respondents ranked the two top reasons why Cloud SLAs are important as (1) being able "to better understand the level of security and data protection offered by the CSP" (41 percent), and (2) "to monitor the CSP's performance and security levels" (35 percent). Furthermore, based on the respondents' experiences, the key issues needed to make Cloud SLAs "more usable" for Cloud Customers highlighted: (1) the need for "clear SLO metrics and measurements" in first place (66 percent); (2) "making the SLA's easy to understand for different audiences (managers, technical legal staff, etc.)" in second place (62 percent); (3) "having common/standardized vocabularies" (58 percent) in third place; and (4) "clear notions of/maturity of SLAs for Security" (52 percent) in fourth place. These responses are empirical indicators of the need to develop the field of Cloud secSLAs, and the techniques to reason about them.

3.2 Which Elements Comprise a secSLA?

A Cloud SLA is a documented agreement between the CSP and the Customer that identifies Cloud services and service level objectives (SLOs), which are the targets for service levels that the CSP agrees to meet. If a SLO defined in the Cloud SLA is not met, the Cloud Customer may request a remedy (e.g., financial compensation). If the SLOs cannot be (quantitatively) evaluated, then it is not possible for Customers or CSPs to assess if the agreed SLA is being fulfilled. This is particularly critical in the case of secSLAs, but it is also an open challenge on *how to define useful (and quantifiable) security SLOs?*

In general, a SLO is composed of one or more metrics (either quantitative or qualitative), where the SLO metrics

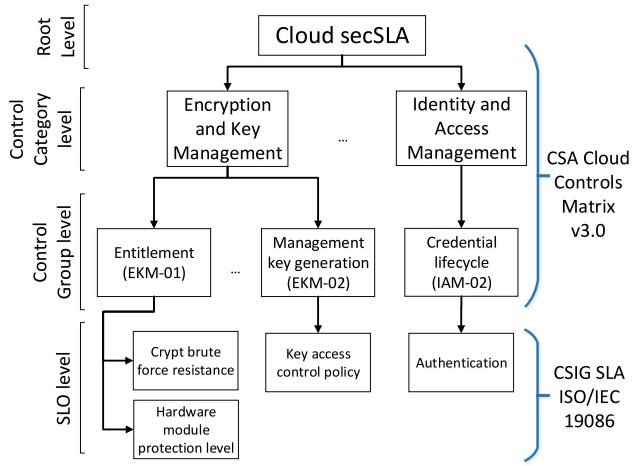


Fig. 1. The Cloud secSLA hierarchy.

are used to set the boundaries and margins of errors CSPs have to abide by (along with their limitations). Considering factors such as the advocated familiarity of practitioners with security controls frameworks(e.g., ISO/IEC 27002 [1], the Cloud Security Alliance's Cloud Control Matrix [2], and the National Institute of Standards and Technology SP 800-53 [3]), the relevant workgroups (e.g., the EC's Cloud Select Industry Group on Service-Level Agreements C-SIG SLA in [7]) have proposed an approach that iteratively refines individual controls into one of more measurable security SLOs. The elicited SLOs metrics can then be mapped into a conceptual model (such as the one proposed by the members of the NIST Public RATAX Working Group [26]), in order to fully define them.

Based on our analysis of the state of practice, Cloud secSLAs are typically modelled using the hierarchical structure shown in Fig. 1. The root of the structure defines the main container for the secSLA. The second and third levels represent the Control Category and Control Group respectively, and they are the main link to the security framework used by the CSP. The lowest level in the secSLA structure represents the actual SLOs committed by the CSP, which threshold values are specified in terms of security metrics. Draft standards like ISO/IEC 19086 and published technical reports like C-SIG SLA and NIST RATAX consider the compositional nature of security SLOs/metrics (introducing terms like Concrete Metrics [26] and Components [23]), although there is still a conspicuous lack of real-world secSLA CSP data compliant with these documents. To the best of our knowledge none of the surveyed standards/best practices, repositories and secSLAs discusses the fact that secSLA elements might have dependencies among them (e.g., representing the trade-offs between performance and security).

For example in Fig. 1, let us suppose that a CSP implements the secSLA Control "Entitlement (i.e., EKM-01)" from the CSA CCM.[2] As observed in the figure, this control is actually contained within the group "Encryption and Key Management (i.e., EKM)". After selecting EKM-01, the same CSP then refers to the SLO list provided on the C-SIG SLA

1. https://cloudsecurityalliance.org/events/securecloud2014/

2. The full implementation/procedures associated to security controls are usually specified by CSPs on their security certifications.

TABLE 1
Excerpt from a Real secSLA Hierarchy

Control Category	Control Group	SLO	Metric	Description	Potential Values
Identity	Access Management	User authentication level	Use of client certificates	Enables client certificates for SSL/TLS	Required $>$ Preferred $>$ Forbidden
			FIPS compliance support	Describes FIPS compliance support	Yes $>$ No
Encryption and Key Management	Entitlement	Cryptographic Strength	Encryption algorithm key length	Data at rest encryption key length	$112 < 128 < 256 < 512$
		Hardware module's protection level	HW Security Level	Security level of hardware modules	$L_0 < L_1 < L_2 < L_3$
	Management key generation	Key access control policy	Client-side encryption level	Describes cryptographic key protections	$L_0 < L_1 < L_2 < L_3$

report [7] (or any other relevant standard) and finds out that two different SLOs are associated with control EKM-01, i.e., "Cryptographic brute Force Resistance" and "Hardware module protection level". Both SLOs are then refined by the CSP into of one of more security metrics, which are then specified as part of the secSLA offered to the Cloud Customer. For example, a CSP can commit to a "Cryptographic brute Force Resistance" measured through security levels such as $(level_1, \ldots, level_8)$, or through a metric called "FIPS compliance" defined as boolean YES/NO values. Therefore, the secSLA could specify two SLOs: (Cryptographic brute Force Resistance $= level_4$), and (FIPS compliance $= YES$). If any of these committed values is not fulfilled by the CSP, then the secSLA is violated and the Customer might receive some compensation (this is the so-called secSLA remediation process).

Table 1 shows the full secSLA hierarchy from Fig. 1, along with a brief summary of the metrics associated to each SLO. As part of the research presented in this paper, industrial and academic volunteers of CSA are developing a catalogue of Cloud security metrics for the purpose of making the presented refinement process easier to implement by stakeholders.[3]

Using the presented approach, the security SLOs proposed by the CSP can be matched to the Cloud Customer's requirements before acquiring a Cloud service. Actually, these SLOs provide a common semantic that both Customers and CSP's can ultimately use to automatically negotiate Cloud secSLAs (cf., Section 5). As a note, the process presented in this section to elicit security SLOs (that will become part of the CSP's secSLA) was adopted by the European project SPECS (Secure Provisioning of Cloud Services based on SLA Management[4]), and is being also used by standardisation bodies such as ISO/IEC and industrial working groups as C-SIG SLA [7].

3.2.1 Considering Dependencies

In real-world Cloud scenarios, the process described in this section should take into account that most Cloud services have horizontal (Cloud supply chains) and vertical (e.g., different Cloud service model layers) dependencies. Thus, it does not suffice to understand how the Cloud service under one unique CSP's control may affect its own Customers, but one also needs to consider how the subservices/CSPs contribute to the overall security level. Hence, there is a distinct need for aggregation of security metrics guaranteed by individual Cloud services in order to get the values for a composite one. While practitioners have acknowledged the challenges associated with the composition of security metrics long before the "Cloud times" [27], nowadays this topic is still mostly unexplored in Cloud systems.

Other relationships commonly appears in relevant standards and best practices, where metrics are *directly depending* in order to allow their composition to generate more complex ones. For example, we can assume direct dependencies at the metric level in Table 1, so that the "Client-side encryption Security Level" results from the composition of both "HW Security Level" and "Encryption algorithm key length" metric. As mentioned above, the Cloud Service Metrics model from NIST [26] supports these direct dependencies through the notion of Concrete and Abstract metrics.

Due to the lack of empirical data to model and validate horizontal and vertical dependencies, the rest of this paper will only consider the compositional nature of security metrics and associated SLOs (direct dependencies) just as discussed by the relevant standards and best practices presented in this section. On this background, the following section presents two approaches to aggregate and evaluate Cloud security levels based on secSLAs.

4 Quantitative Assessment of Cloud secSLAs

The quantitative security-level assessment of CSPs based on secSLAs (for their match to the Customer requirements) is the primary objective of the techniques developed in this section, namely the Quantitative Policy Trees [8] and the Quantitative Hierarchy Process (QHP) [9]. Using this assessment, the CSPs are ranked (as per their secSLAs) for the best match to the Customer requirements. QPT utilizes a logical aggregation of security quantifiers, while QHP is based on multi-variable optimization techniques considering the various elements of a secSLA (as presented in Section 3.2) as the optimization criteria. We first detail the standalone operations of each technique from the secSLA perspective, and subsequently discuss guidance for their usage discretely and collectively. Also the empirical validation of these techniques will be presented through two use cases (using

3. This catalogue is still work in progress, but interested parties can contact the corresponding author.

4. http://www.specs-project.eu

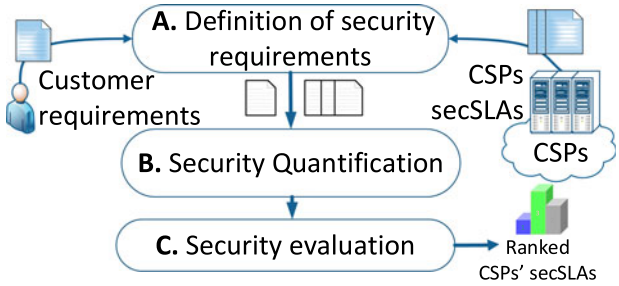


Fig. 2. Stages comprising the quantitative secSLA assessment.

real-world data) in Section 5. As an overview of the two techniques, the secSLA assessment and the ranking of CSPs is performed in progressive stages (common to both QPT and QHP techniques), as shown in Fig. 2.

In Stage (A), we express in a common way both the Customer's security requirements and the CSP's committed SLOs using a standardised secSLA template (e.g., based on ISO/IEC 19086 [23]. In Stage (B), the Customer's requirements and CSP's secSLA are quantitatively evaluated. This quantitative data is then used in Stage (C) as input to a ranking algorithm, in order to provide the final assessment result. We detail each of the two techniques (QPT and QHP) in the subsequent sections.

4.1 Quantitative Policy Trees

Luna et. al. [8] proposed the use of a tree-like data structure (i.e., the Quantitative Policy Tree), to model a CSP's security policy in order to numerically evaluate it with respect to a set of Customer's requirements. While the original QPT was designed to evaluate security control frameworks such as CSA CCM [2], this section develops an extended QPT approach for the quantitative evaluation of Cloud secSLAs.

4.1.1 Stage A. Definition of Security Requirements

The QPT is an AND/OR tree[5] where the Cloud Customer's requirements are represented also as a security SLA (called *User secSLA*). The Control Categories, and Controls are represented as intermediate nodes of the tree, while security metrics associated to SLOS are represented as *weighted* leaf nodes. Assigned weights are used to represent the relative importance of SLOs from the Customer's perspective (e.g., for some users the SLO metric "Encryption Key Size" might be more important than SLO metric "Backup Frequency"). The basic rules for setting weights on the *User secSLA's* individual security SLOs are:

- Each Customer required security SLO will be associated with a quantitative weight ω_i $(0 \leq \omega_i \leq 1)$.
- The sum of all the weights ω_i associated with a set of sibling security SLO metrics (i.e., those having the same parent Control) must be equal to 1.
- The Customer can choose specific elements of the secSLA (cf., Section 3) to benchmark by assigning $\omega_i = 0$ to those not of interest.

5. While we use a binary tree and the basic AND/OR operations, the concept is directly extensible to an X-ary tree and for complex logical operations.

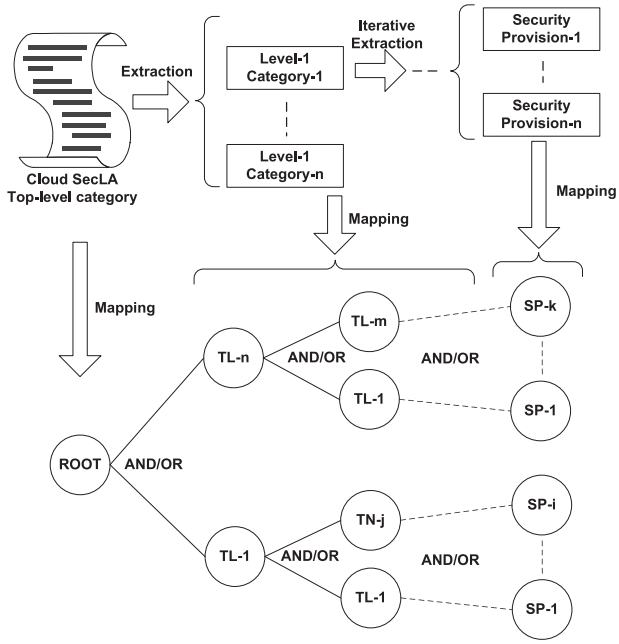


Fig. 3. secSLA-to-QPT: Mapping a Cloud SecLA into a QPT[8].

To complete the customization of a *User secSLA*, the Customer can also select the appropriate AND/OR relationships[6] between the different Control Categories, Controls and , SLO metrics of the secSLA. As inferred from their name, AND relationships will model *hard-requirements* where "Categories A, B and C are *all* required due to regulatory compliance", whereas OR relationships are more adequate to model *soft-requirements* e.g., "Either A, B or C are needed to achieve my security goals". The overall QPT creation process is shown in Fig. 3.

4.1.2 Stage B. Security Quantification

In order to evaluate the *User secSLA* (termed as the User QPT) with respect to the offered *CSP secSLA* (CSP QPT), the QPT utilizes the notion of local security levels (LSL) [22] and two basic assumptions: *(i)* all the *i-leaf nodes* on the QPT have been already associated with a $LSL_i > 0$ and, *(ii)* there exists a maximum value LSL_{max} that is the same for *all* the leaf nodes of the QPT (i.e., $0 < LSL_i \leq LSLmax$).

Once each leaf node in the QPT has been associated with the duple $\{LSL_i, \omega_i\}$, it is possible to propagate these values to the rest of the tree using the aggregation rules shown in Table 2. Notice that QPT's AND/OR relationships allow modelling metrics/SLOs with direct dependencies, where low-level metrics (i.e., Abstract metrics according to NIST [26]) can be composed into more advanced/high-level ones (i.e., Concrete metrics [26]).

4.1.3 Stage C. Security Evaluation

Once the *User QPT* and the *CSP QPT* have been populated with the aggregated values (quantitatively computed from

6. As mentioned in Footnote 5, multi-level aggregations, correlations and complex logical operators are possible as per the needs of the security characterization. We have limited the presentation for ease of presentation of the concept to the basic case of binary tree with AND/OR operations. For complex logics, the aggregation rules of Table 2 need to be extended as needed for the desired logical composition.

TABLE 2
Aggregation Rules for a QPT with *n*-Sibling Nodes [8]

	Aggregation rule with $i = 1 \dots n$	
Parameter	AND node	OR node
$Agg_{ParentL1}$	$\sum_{i=1}^{n}(LSL_i \times \omega_i)$	$min(LSL_i \times \omega_i)$
$Agg_{ParentL2}$	$\sum_{i=1}^{n} Agg_{ParentL1,i}$	$min(Agg_{ParentL1,i})$

Table 2), it is possible to apply a ranking process to determine how different CSPs under-/over-provision a Customer's requirement. Luna et. al. [8] proposes two different classes of benchmarks, namely $QuantB_{node}$ (cf. Definition 1) and $QualB_{node}$ (cf. Definition 2), both based on the quantitative security values already aggregated in the QPT.

Definition 1. *The quantitative benchmark $QuantB_{node}$ associated with a specific* node *of the QPT, is defined as follows:*

$$QuantB_{node} = \frac{Agg_{CSP,node} - Agg_{User,node}}{Agg_{max,node}}$$

Where:

- *$Agg_{CSP,node}$ is the aggregated security value for* node *in the* CSP QPT*, as computed with Table 2.*
- *$Agg_{User,node}$ is the aggregated security value for* node *in the* User QPT*, as computed with Table 2.*
- *$Agg_{max,node}$ is is the aggregated security value for* node *in* either User QPT *or* CSP QPT*, as computed with Table 2 and using the maximum Local Security Level (LSL_{max}).*

Definition 2. *The following expression defines $QualB_{node}$, the qualitative benchmark associated with a specific* node *of the QPT:*

$$QualB_{node} = \begin{cases} \lceil QuantB_{node} \times Ranks_{max} \rceil & if\ QuantB_{node} \geq 0 \\ \lfloor QuantB_{node} \times Ranks_{max} \rfloor & if\ QuantB_{node} < 0, \end{cases}$$

where:

- *$QuantB_{node}$ is the quantitative benchmark in Definition 1.*
- *$Ranks_{max}$ is the total number of chosen qualitative labels minus one. For example, if the set of qualitative labels is* {"Copper", "Silver", "Gold"} *then $Ranks_{max} = 2$*

The result of the previous QPT metric is an integer number such that $QualB_{node} = \{-Ranks_{max}, \dots, 0, \dots, Ranks_{max}\}$. In order to assign it a qualitative label from the set $Ranks = \{Label_1, \dots, Label_n\}$ where $n = Ranks_{max} + 1$, we use the following mapping function:

$$f(QualB_{node} \mapsto Ranks) = \begin{cases} Label_1 & \text{if } QualB_{node} = 0 \\ Label_2 & \text{if } QualB_{node} = 1 \\ -Label_2 & \text{if } QualB_{node} = -1 \\ \quad \vdots & \\ Label_n & \text{if} QualB_{node} = Rank_{max} \\ -Label_n & \text{if} QualB_{node} = -Rank_{max}. \end{cases}$$

In the previous function, notice that a "negative" label such as $-Label_n$ literally represents the counterpart of the corresponding "positive" label $Label_n$. For example "A-" and "A+", "Silver" and "Silver-", and so forth.

4.2 Quantitative Hierarchy Process

The quantitative security assessment of CSP's using control frameworks is the primary objective of the Quantitative Hierarchy Process, as originally introduced in [9]. By applying the QHP assessment technique, the CSPs can be ranked (as per their offered security controls) depending on how well they match the Customer's requirements. QHP allows Cloud Customers to (i) compare, benchmark and rank the aggregated security level provided by two or more CSPs, (ii) provide a composite quantitative and qualitative security assessment technique based on the well-known AHP [28] (depending on the user defined security requirements and priorities), (iii) allow Customers with different levels of security expertise to specify their security requirements at varied levels of granularity, and (iv) automate the overall assessment process.

Similar to the QPT, the secSLA assessment and ranking of CSPs is a proposed extension of the original QHP [9], as developed in the following progressive stages:

4.2.1 Stage A. Definition of Security Requirements

In this stage, the Customer creates its set of security requirements based on the same secSLA template (structure) used by the CSPs to specify their security offers. The secSLA template will have the structure presented in Section 3 (i.e., from Control Categories to individual security metrics associated to committed SLOs).

The Customer-defined requirements are distinctive elements of a Cloud secSLA, where all the elements are weighted or evaluated in order to represent their relative importance from the Customer's perspective. For example, the (prospective) Cloud Customer might specify that some specific Control is "Very Important", or even request a specific key length value for an "Encryption Key" SLO metric. The output of this stage will be a set of Customer security requirements specified as a secSLA.

4.2.2 Stage B. Security Quantification

In order to evaluate the Customer requirements with respect to a CSP secSLA, the so-called measurement model for different security SLO metrics needs to be defined. Over this stage, different comparison metrics for different types of requirements are defined, so they can be applied for the quantitative security assessment. The terms shown in Table 3 are used to present the QHP framework.

Definition 3. *The relationship across the CSP's with respect to security SLO (V) is represented as a ratio:*

$$CSP_1/CSP_2 = V_1/V_2.$$

The security SLOs metrics under evaluation can be boolean (e.g., a YES/NO representing the need of a security mechanism) or numbers (e.g., a cryptographic key length) such that:

- *Boolean:* In this case the CSP's YES/NO SLO's metric values are defined as boolean $true$ and $false$ or 1 and 0, respectively. The relationship across the CSP's

TABLE 3
Used Terms Definitions

Term	Definition
k	security metric associated to the SLO.
CSP_i	Cloud provider i, such that $i \in \{1, \ldots, n\}$, where n is the total number of CSPs.
V_i	SLO value for based on metric k, and provided by CSP_i (CSP_i provides k with value V_i).
CSU	Cloud Service Customer.
V_{csu}	Customer requested value for SLO metric k.
W	relative rank ratio.
CSP_1/CSP_2	indicates the relative rank W of CSP_1 over CSP_2, regarding k. Or relative rank $1/W$ of CSP_2 over CSP_1, regarding k.
CSP_i/CSU	indicates the relative rank of CSP_i over CSU, which specifies if CSP_i satisfies CSU requirements, with respect to k.

with respect to security SLO metric value (V) based on Definition 3 can be represented as:

$$\begin{aligned} CSP_1/CSP_2 &= 1 \quad if \quad V_1 = 1 \\ &= 0 \quad if \quad V_1 = 0. \end{aligned}$$

- *Numerical:* Assume e.g., a cryptographic key length (in bits) defined as k and specified by $\{64, 128, 256, 512, 1{,}024, 2{,}048\}$, such that $64 < 128 < 256 < 512 < 1{,}028 < 2{,}048$, which is defined as $level_1, level_2, level_3, level_4, level_5, level_6$. The security levels are modelled as $\{1, 2, 3, 4, 5, 6\}$ respectively, such that $1 < 2 < 3 < 4 < 5 < 6$. Thus, the relationship across the CSP's with respect to security SLO value (V) based on Definition 3 can be represented as:

$$\begin{aligned} CSP_1/CSP_2 &= 1 \quad if \quad V_1 \equiv V_2 \\ &= W \quad if \quad V_1 > V_2 \\ &= 1/W \quad if \quad V_1 < V_2. \end{aligned}$$

The resulting value can be interpreted in two different ways: higher is better (e.g., encryption key size) or lower is better (e.g., backup frequency). If higher is better then V_1/V_2 is the value of CSP_1/CSP_2 and if lower is better then V_2/V_1 is the value of CSP_1/CSP_2.

4.2.3 Stage C. Security Evaluation

Given the fact that a secSLA might have a high number of individual security SLOs and that Customers might specify their requirements with different levels of granularity, the challenge is not only how to quantify different metrics associated to these SLOs, but also to aggregate them in a meaningful way. To solve these challenges, QHP's ranking mechanism is based on AHP [28] for solving Multiple Criteria Decision Making (MCDM) [29] problems.

The AHP-based methodology for CSP rankings consists of four main steps: (1) hierarchy structure (2) weights assignment (3) pairwise comparison and (4) attributes aggregation to give the overall rank calculation. These steps are summarised next:

4.2.4 Hierarchy Structure

The secSLA's are modelled as a hierarchical structure (cf., Fig. 1), such that the top-most layer of the hierarchy structure defines the main goal and aims to find the overall rank (i.e., the root "SLA-level"). The lowest level is represented by the actual security metrics related to the committed SLO value.

4.2.5 Weights Assignment

Customer-defined weights are assigned to the different levels of the secSLA hierarchy to take into account their relative importance. QHP considers two types of weights:

- *User assigned qualitative values*. Customers assign the desired weights to each SLO metric to indicate their priorities (High-Important (HI), Medium-Important (MI), Low-Important (LI)). These labels are transformed to quantitative values and assigned as normalized numbers to satisfy the AHP requirements.
- *Using AHP's standard method*. The Customer can assign numeric weights to each one of the secSLA elements using values in some defined scale. For example, the AHP method proposes a scale from 1 to 9 to indicate the importance of one element over another.

4.2.6 Pairwise Comparison

In this phase, the relative ranking model defining the most important requirements and their quantitative metrics is specified. This ranking model is based on a pairwise comparison matrix of secSLA elements provided by different CSPs as required by the Customers. Using a Comparison Matrix (CM) for each CSP, a one-to-one comparison of each CSP for a particular attribute is obtained, where C_1/C_2 indicates the relative rank of C_1 over C_2. This will result in a one to one comparison matrix of size n x n (if there are a total of n CSP's), such that:

$$CM = \begin{array}{c} \\ CSP_1 \\ CSP_2 \\ \vdots \\ CSP_n \end{array} \begin{array}{c} \begin{array}{cccc} CSP_1 & CSP_2 & \ldots & CSP_n \end{array} \\ \begin{pmatrix} CSP_1/CSP_1 & CSP_1/CSP_2 & \ldots & CSP_1/CSP_n \\ CSP_2/CSP_1 & CSP_2/CSP_2 & \ldots & CSP_2/CSP_n \\ \vdots & \vdots & \ddots & \vdots \\ CSP_n/CSP_1 & CSP_n/CSP_2 & \ldots & CSP_n/CSP_n \end{pmatrix} \end{array}. \tag{1}$$

The relative ranking of all the CSPs for a particular SLO metric is given by the eigenvector of the comparison matrix. This eigenvector shows a numerical ranking of CSP's that indicates an order of preference among them as indicated by the ratios of the numerical values, which is called Priority Vector (PV).

4.2.7 Attributes Aggregation

In the final phase, the assessment of the overall security level (and consequently the final ranking of CSPs) is obtained using a bottom-up aggregation. To achieve that, the PV of each attribute is aggregated with their relative weights assigned in Step 2. This aggregation process is repeated for all the attributes in the hierarchy along with their relative weights.

$$PV_{aggregated} = (\, PV_1 \quad \ldots \quad PV_n \,)(\, w_i \,), \tag{2}$$

where w_i is a Cloud Customer assigned weight for criteria i.

TABLE 4
QPT and QHP—Comparison of Main Features

Stage	Feature	Evaluation technique	
		QPT	QHP
Security Requirements	secSLA granularity for expressing requirements	Weights and values only at SLO level	Weights and values at all levels
	Supported SLO values	Quantitative and Qualitative	
	Template for Customer requirements	secSLA hierarchy	
	Model relationships among secSLA elements	AND/OR among SLOs	None
Security Quantification	Base technique for aggregation	Ad-hoc	Multi-criteria decision technique
	Used secSLA abstraction	AND/OR Tree	Matrix
Security Evaluation	Output	Ranked List, Overall Security Level	
	Format of resulting security level	Quantitative/Qualitative	Quantitative

4.3 QPT and QHP Comparison

Table 4 summarises the main features found in both the QPT and QHP methodologies presented in this section. The empirical validation presented in the following section, will complement the features shown in Table 4 with a set of usage guidelines based on real-world use cases. In this section, our focus is to introduce a set of criteria aiming to guide early QPT/QHP adopters in aspects related to the requirements of their specific application scenarios:

- As the QPT aggregation is based on AND/OR operations, and as the CSPs ranking (with respect to Customer requirements) is only executed at the root (highest) level, it clearly has the potential to outperform QHP's aggregation time. In QHP, the CSPs ranking is performed at each level of the secSLA hierarchy structure, which means that by increasing the number of SLOs QPT shows better performance regarding aggregation time. This might be a useful feature in scenarios where low-latency is needed e.g., Infrastructure-as-a-Service (IaaS) scheduling, and automation. Section 5 will empirically demonstrate this assertion.
- QHP's ability to depict CSPs ranking at each level of the secSLA hierarchy gives both CSPs and Customers the ability to determine which security SLOs are over/under provisioning the Customer's requirements. This is useful for CSPs to improve their provided secSLAs match to the Customer's requirements.
- QHP's flexibility to represent Customer requirements at different levels of secSLA hierarchy (i.e., from Control Category to individual SLO metrics), makes it more "user-friendly" and suitable for implementations where human-interaction is needed e.g., in a decision making dashboard (cf. Section 6). QPT can only evaluate security requirements specified at the SLO-level, thus it is better suited for scenarios where Customers can express security preferences at a very granular level (e.g., software agents negotiating secSLAs).
- QPT and QHP can also be used complementarily. For example, QHP can be used by prospective Customers manually exploring different CSP's offers through what-if scenarios. Once a secSLA has been agreed upon, then applications can rely on QPT for dynamically negotiating new terms without Customer intervention.
- QHP relies on a mature set of techniques (i.e., multi-criteria decision analysis or MCDA), which eases its extensibility to add new features with few efforts. For example, the use fuzzy MCDA techniques is part of our future work to add the notion of uncertainty to the security evaluation process.

The next section empirically demonstrates the features of both QPT and QHP based on two use case scenarios.

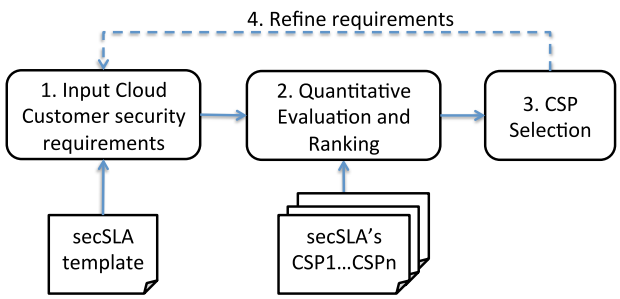


Fig. 4. Selecting a CSP based on its secSLA.

5 QPT and QHP Validation: Case Studies

This section has two main objectives (a) empirical validation of QPT/QHP, and (b) demonstrating the advantages and disadvantages of each approach.

The empirical validation is performed through two scenarios that use real world secSLAs structured in compliance with the current draft version of the ISO/IEC 19086 standard [23]. and with data derived from the Cloud Security Alliance's STAR repository [30]. The associated metrics were extracted from the CSA metrics catalog referenced in Section 3.2. By following the refinement approach shown in Fig. 1 and presented in Section 3, our validation approach created a dataset comprised of three Cloud secSLAs[7] that were chosen to cover all possible conditions for each SLO (i.e., over/under provisioning or satisfying the Cloud Customer's requirements). Each secSLA contained an overall of 139 SLOs (with both quantitative and qualitative metrics), and with real values corresponding to the CSP information found on the CSA STAR repository.

Fig. 4 shows the process used to systematically perform the CSP comparison presented in the rest of this section. The overall process consists of four steps (common to both QPT and QHP techniques presented in Section 4) namely:

7. For confidentially reasons, the name of the CSPs have been anonymised.

TABLE 5
Case Study 1: Excerpt of CSP's secSLAs and Customer Requirements

Cloud secSLA Element based on CSA STAR [30]			CSP_1	CSP_2	CSP_3	C_i		
Control Category	Control Group	SLO	Val_1	Val_2	Val_3	Case I	Case II	Case III
Compliance (CO)	audit planing (CO1)	CO1.1	*yes*	*yes*	*yes*	*yes*	*High*	*High*
		CO1.2	$level_3$	$level_2$	$level_3$	$level_3$		
	independent audits (CO2)	CO2.1	*no*	*yes*	*yes*	*yes*	*Low*	
		CO2.2	*yes*	*yes*	*yes*	*yes*		
		CO2.3	*yes*	*yes*	*yes*	*yes*		
		CO2.4	*yes*	*yes*	*yes*	*yes*		
	Third party audits (CO3)	CO3.1	*yes*	*yes*	*yes*	*yes*	*Medium*	
		CO3.2	*yes*	*yes*	*yes*	*yes*		
		CO3.3	*Quarterly*	*Annual*	*Monthly*	*Monthly*		
Facility Security (FS)	Secure Area (FS1)	FS1.1	*no*	*yes*	*yes*	*yes*	*High*	*Low*
		FS1.2	*yes*	*no*	*yes*	*yes*		
	Asset Management (FS2)	FS2.1	*yes*	*yes*	*yes*	*yes*		
		FS2.2	$level_3$	$level_2$	$level_3$	$level_3$		
		FS2.3	*yes*	*yes*	*yes*	*yes*		
Risk Management (RI)	Risk assessments (RI1)	RI1.1	*Internal*	*Internal*	*External*	*Internal*	*Internal*	*Medium*
		RI1.2	*yes*	*yes*	*yes*	*yes*	*no*	

1) Step 1. Cloud Customer security requirements: In this step the (prospective) Cloud Customer defines his security requirements (SLO's thresholds and associated weights), and express them using a standardised secSLA template (e.g., based on ISO/IEC 19086 [23]).
2) Step 2. Quantitative Evaluation and Ranking: The Customer's security requirements (Step 1) are evaluated with respect to the CSP's secSLAs. As shown in Section 5.1, QPT and QHP have different capabilities and the decision on which one to use will mainly depend on the Cloud Customer's degree of security expertise.
3) Step 3. CSP Selection: The output from Step 2 is a set of CSPs ranked with respect to the Customer's Security Requirements. In this step, any of the CSPs should be selected by the Customer, otherwise the whole process might be repeated with a refined set of security requirements (Step 4).
4) Step 4. Refine Requirements: This step is used in case the Cloud Customer decides to change his security requirements (e.g., with new weights assigned to selected SLOs) and repeat once again the whole comparison process, as shown in Section 5.2.

Our validation scenarios were designed taking into account real concerns from Cloud Customers (i.e., procuring Cloud services based on security requirements) and CSPs (i.e., maximising offered security levels). The used data set also consisted of different combinations of requirements and real secSLA representing three different Customers (as shown in Table 5), and three different CSPs respectively.

It is important to notice that in compliance with the ISO/IEC 19086 standard [23], the dataset used for our experiments only contained secSLAs which elements (controls, SLOs) are independent but keep their compositional nature. Using terminology from this standard, the compositional nature of the secSLA is based on top-level components (e.g., cryptography) comprising one of more measurable service commitments (e.g., cryptographic access control policy, key management, and data at rest). Both assumptions (lack of dependencies, compositional nature) are also consistent with the C-SIG SLA guidelines [7]), and the NIST Cloud Service Metrics model [26].

5.1 The Customer Perspective: Security Comparison of CSPs

This initial validation scenario demonstrates how a (prospective) Cloud Customer can apply the techniques presented in Section 4 to compare side-by-side three different CSPs based on their advertised secSLAs, and with respect to a particular set of security requirements (also expressed as a secSLA). Section 6 will present the prototype of a decision making dashboard that automates the comparison task.

Table 5 presents a sample dataset used for this scenario, where based on the information available in the CSA STAR repository [30], the values associated to 16 SLO metrics (out of 139) for the three selected CSPs are presented. In order to perform a comprehensive validation, the selected SLOs comprised both qualitative (e.g., YES/NO) and quantitative (e.g., security levels from 1 to 4) metrics. The "YES/NO" SLO's thresholds are modelled as boolean $1/0$, whereas SLOs associated to security levels as $level_1, level_2, level_3, level_4$ are modelled as $\{1, 2, 3, 4\}$. For example, the *CO3.3* SLO is defined using qualitative thresholds (None, Annually, Quarterly, Monthly) which are specified as $level_1, level_2, level_3, level_4$. Similarly, the *RI1.1* SLO is defined using qualitative (Internal, External) values.

Furthermore, Table 5 also shows three sets of Cloud Customer requirements used as baseline for comparing the selected CSPs. For validation purposes the Customer (Cloud Service User (CSU)) requirements are being expressed at different levels of granularity (as mentioned in Section 4):

- In column "Case I", Customer requirements are expressed at a very granular level (i.e., per-SLO). This represents a security-expert user.
- Column "Case II" shows a set of requirements expressed at three different levels of granularity

TABLE 6
Absolute Quantitative Benchmarks Obtained for Three Different CSP's secSLAs

	CSP_1	CSP_2	CSP_3
secSLA	0.85	0.83	1
CO	0.86	0.89	1
FS	0.8	0.7	1
RI	1	1	1

(corresponding to the hierarchy shown in Fig. 1) namely SLO, Control Group, and Control Category. Notice that at the Control Group and the Control Category level, the Customer expresses his requirements depending on the relative importance[8] of the SLA element (e.g., high, medium, or low).

- Finally, in column "Case III" are shown Customer requirements only at the Control Group and Control Category levels. This might be the case of a user that is not security expert.

In order to evaluate the CSP's secSLAs with respect to the Customer requirements we proceed to apply the techniques presented in Section 4 (cf., Step 2 in Fig. 4).

5.1.1 Case I: An Expert Customer

The quantitative evaluation of the Cloud security SLOs defined in Table 5 regarding Customer Case I is detailed in this section.

Using the QPT

For comparison purposes, all the QPT analyses shown in this section considered (i) a maximum of 4 Local Security Levels (i.e., LSL = 4), (ii) all leaf nodes on the QPT having the same weight, (iii) only AND relationships on the QPT, (iv) YES/NO values specified as LSL_{max} and LSL_{min} respectively (i.e., $LSL_{max} = 4$ and $LSL_{min} = 0$), and finally (v) security levels specified using LSLs from 1 to 4.

For QPT we performed two sets of evaluations, first with the three CSPs to show individually which CSP outperforms the other two. Then, evaluating the three CSPs with respect to the Customer (CSU) requirements.

Table 6 shows the CSPs secSLA aggregation using the rules specified before in Table 2. The information shown in Table 6 is useful to analyse how individual Control Categories contribute to the overall security level of the CSP. For example, if control *CO* is the prime requisite from a business perspective, then the absolute evaluation will advise to initially choose CSP_3 followed by CSP_2 over CSP_1. Notice that this conclusion cannot be drawn directly from the overall secSLA level benchmarks, where CSP_1 outperforms CSP_2.

A second set of benchmarks was applied to the dataset of the three Cloud secSLA regarding the Customer secSLA requirements. Definition 1 is used to show the quantitative benchmark $QuantB_{node}$ associated with each *node* of the QPT as shown in Table 7. For example, CSP_1 is under-provisioning *CO2*, *CO3* and *FS1*. While CSP_2 is not fullfiling the Customer requirements for *CO1*, *CO3*, *FS1* and *FS2*. Only CSP_3 fulfils the Customer requirements as shown in overall secSLA rank. The aggregated secSLAs values are normalized with respect to the Customer requirement (cf., Fig. 5).

8. This is the typical result of a risk assessment.

TABLE 7
Quantitative Benchmarks Obtained for Three Different CSP's secSLAs Based on Customer's Case I Requirements

	CSP_1	CSP_2	CSP_3
secSLA	−0.176	−0.2	0
CO1	0	−0.33	0
CO2	−0.33	0	0
CO3	−0.1	−0.22	0
CO	−0.167	−0.129	0
FS1	−1	−1	0
FS2	0	−0.2	0
FS	−0.25	−0.43	0
RI	0	0	0

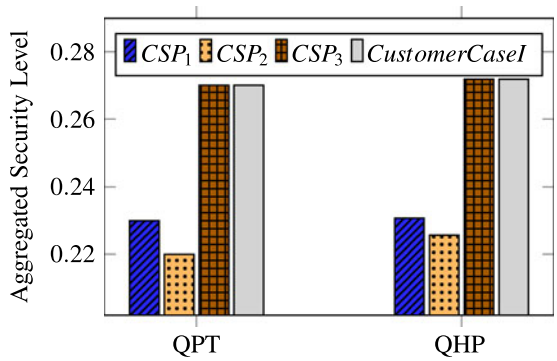


Fig. 5. Comparing QPT and QHP for customer case I requirements.

Using the QHP

For this evaluation technique, the Customer specifies his requirements at the lowest level of the secSLA (i.e., SLOs) and considers the same relative importance (i.e., weights) for all of these. Prior to the calculation of the relative ranking matrix using Equation (1), the following considerations take place:

1) QHP uses qualitative weights to indicate the Customer's relative priorities, and these weights are normalized as to comply with AHP requirements.
2) All SLOs specified by the Customer as boolean NO, are assigned a relative rank value 0.
3) All SLOs specified by the Customer as boolean YES, are assigned a relative rank value 1.
4) *High-Important* and *Low-Important* indicate a weight 1 and 0 respectively.
5) *Medium-Important* can be considered any intermediate values between 1 and 0. In this analysis *Medium-Important* indicates a weight 0.5.
6) All CSPs security SLOs are normalized to the Customer requirements to eliminate masquerading.[9]

For the Compliance Control Category, there are three security Control Groups which are further divided into a set of SLOs (as shown in Table 5). Definition 3 is used to create the attribute pairwise relation, as for example in the case of *CO1.2*:

$$CSP_1/CSP_2 = 3/2 \quad CSP_2/CSP_3 = 2/3$$
$$CSP_3/CSP_1 = 3/3 \quad CSU/CSP_2 = 3/2.$$

9. The masquerading effect happens when the overall aggregated security level value mostly depend on those security controls with a high-number of SLOs, thus affecting negatively groups with fewer although possibly more critical provisions. Other methodologies for the Cloud security assessment (such as REM [22]) suffer from this effect.

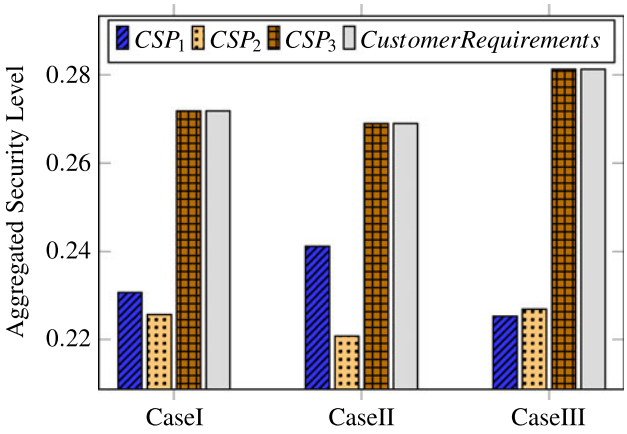


Fig. 6. QHP-based evaluation showing the aggregated secSLA level.

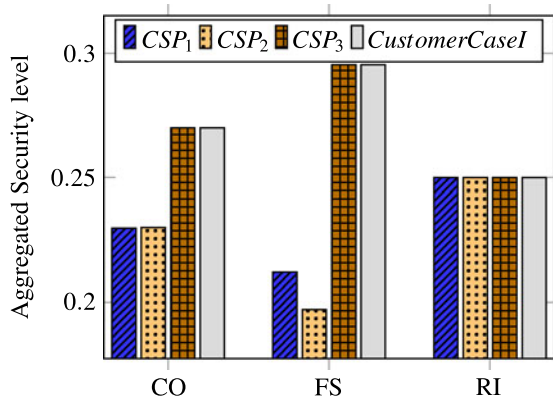


Fig. 7. QHP-based aggregation at the control category-level (for customer case I requirements).

Thus, the CM of *CO1.2* is calculated using Equation (1) such as:

$$CM_{CO1.2} = \begin{array}{c} \\ CSP_1 \\ CSP_2 \\ CSP_3 \\ CSU \end{array} \begin{array}{c} \begin{array}{cccc} CSP_1 & CSP_2 & CSP_3 & CSU \end{array} \\ \begin{pmatrix} 1 & 3/2 & 3/3 & 3/3 \\ 2/3 & 1 & 2/3 & 2/3 \\ 3/3 & 3/2 & 1 & 3/3 \\ 3/3 & 3/2 & 3/3 & 1 \end{pmatrix} \end{array}.$$

The relative ranking of the CSPs for *CO1.2* is given by the priority vector for $CM_{CO1.2}$ ($PV_{CO1.2}$). Similarly, we premeditate $CM_{CO1.1}$ and $PV_{CO1.1}$. PV_{CO1} is then calculated by aggregating $PV_{CO1.1}$ and $PV_{CO1.2}$ with Customer normalized weights (w_{CO1}) using Equation (2). Where PV_{CO1} reflects which of the CSPs provide the *CO1* security SLO relative to other CSPs and to the Customer requirements as shown in Fig. 9, such that:

$$PV_{CO1} = \begin{array}{c} \\ CSP_1 \\ CSP_2 \\ CSP_3 \\ CSU \end{array} \begin{array}{c} \begin{array}{cc} PV_{CO1.1} & PV_{CO1.2} \end{array} \\ \begin{pmatrix} 0.25 & 0.2727 \\ 0.25 & 0.1818 \\ 0.25 & 0.2727 \\ 0.25 & 0.2727 \end{pmatrix} \end{array} \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}.$$

Therefore, PV_{CO1} is:

$$PV_{CO1} = \begin{array}{cccc} CSP_1 & CSP_2 & CSP_3 & CSU \end{array} \\ \begin{pmatrix} 0.2614 & 0.2159 & 0.2614 & 0.2614 \end{pmatrix}.$$

This implies that CSP_1 and CSP_3 equally satisfy CSU's requirement. However, CSP_2 does not fulfill that requirement. The priority vector for *Independent audits* (PV_{CO_2}) is calculated similarly, such that *CO2.1*, *CO2.2*, *CO2.3* and *CO2.4* priority vectors are aggregated. Similarly, we compute PV_{CO3} where *CO3.1*, *CO3.2* and *CO3.3* are specified by the Customer as *Yes*, *Yes* and *Monthly* respectively.

The three *Compliance* priority vectors *CO1*, *CO2*, *CO3* are aggregated to have the overall compliance priority vector PV_{CO} as shown in Fig. 7 such that:

$$PV_{CO} = \begin{array}{cccc} CSP_1 & CSP_2 & CSP_3 & CSU \end{array} \\ \begin{pmatrix} 0.2299 & 0.2301 & 0.27 & 0.27 \end{pmatrix}.$$

Both CSP_1 and CSP_2 under-provision *CO2* and CSP_2 under-provisions *CO1* and *CO3*. As a result, only CSP_3 satisfies *CSU*'s *CO* requirement. In a similar way the *Facility Security* and *Risk Management* priority vectors are considered (Fig. 7).

$$PV_{FS} = \begin{array}{cccc} CSP_1 & CSP_2 & CSP_3 & CSU \end{array} \\ \begin{pmatrix} 0.2121 & 0.1970 & 0.29545 & 0.29545 \end{pmatrix}$$

$$PV_{RI} = \begin{array}{cccc} CSP_1 & CSP_2 & CSP_3 & CSU \end{array} \\ \begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}.$$

Finally, the priority vectors of *Compliance*, *Facility Security* and *Risk Management* security are aggregated to obtain the total secSLA priority vector:

$$PV_{total} = \begin{array}{cccc} CSP_1 & CSP_2 & CSP_3 & CSU \end{array} \\ \begin{pmatrix} 0.2307 & 0.2257 & 0.2718 & 0.2718 \end{pmatrix}.$$

Consequently, only CSP_3 fulfills the Customer's requirements, as shown in Fig. 6.

The proposed framework allows users to visualize the differences between various CSPs with respect to user requirements. Both CSP_1 and CSP_2 under-provisions *CO* and *FS*. As a result, CSP_3 is the best matching provider according to Customer's requirements.

QPT and QHP

Fig. 5 shows the results of applying both QPT and QHP to the set of secSLAs and also the Customer Case I requirements presented in Table 5. As shown in Fig. 5 the resulting ranking of CSP's is consistent for both QPT and QHP: CSP_3 is the provider that better fulfils the customer requirements, followed by CSP_1 and CSP_2 respectively. Where as shown in Fig. 9, CSP_1 is not satisfying user requirements for *CO2.1*, *CO3.3* and *FS1.1* SLOs. Also CSP_2 is not satisfying user requirements for *CO1.2*, *CO3.3*, *FS1.2* and *FS2.2*. For Customers specifying the SLO-level requirements, this means that both techniques result on the same/consistent ranking.

It is worth noting that QPT can only evaluate requirements specified at the SLO-level, therefore it cannot be applied either to Case II or Case III requirements.

The QHP evaluation technique allows Customers to evaluate CSPs security levels and perform comparisons at different levels of granularity. This can be observed in Figs. 7 and 9. Fig. 6 shows the overall security evaluation (i.e., at the top secSLA-level) for each one of the three sets of Customer requirements. Fig. 7 shows a different level of

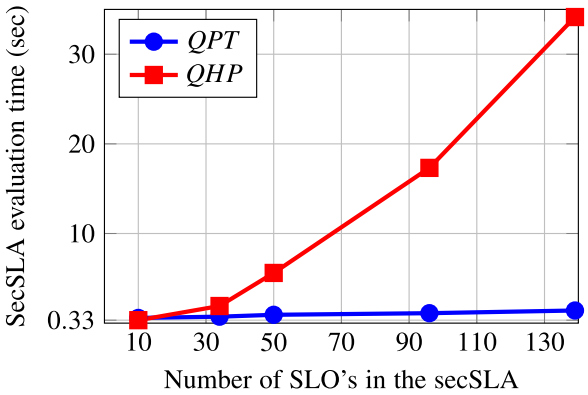


Fig. 8. Performance comparison between QPT and QHP (evaluating customer case I and CSP_1).

aggregation (i.e., Control Category) for the Customer Case I requirements. Fig. 9 shows the CSPs ranking at the SLO-level. For example, during a procurement process Fig. 6 can be used to provide preliminary guidance to select an initial set of CSP's, while a more detailed decision might be based on the more granular Fig. 7 (or even by comparing at the SLO-level as in Fig. 9).

Finally, in Fig. 8 we compare both QPT and QHP from a performance viewpoint. For this experiment we measured the time consumed (in seconds) to evaluate a secSLA comprised of an incremental number of SLOs (up to the 139 contained in our dataset) with respect to Customer Requirements I. It can be observed that in the case of QPT the number of evaluated SLOs does not affects the performance, whereas for QHP the time required to evaluate a secSLA increases exponentially depending on the number of SLOs (as explained in Section 5.1). In scenarios where performance is not important (e.g., decision-making dashboards), then QHP might be used because of the flexibility of showing the evaluations at varied levels of secSLA. However, if secSLA automation is required (e.g., a software agent deciding which Cloud storage provider to use), then QPT would provide the best results from the performance perspective.

5.1.2 Case II: A Semi-Expert Customer

As mentioned in Section 4.2, the QHP technique allows Customers to specify their security requirements at varied levels of granularity. This helps to remove the need for Customers to specify the value required for every single security SLO (which usually needs a extremely high level of expertise). Moreover, allowing Customers to specify their security requirements using qualitative labels, enables both basic and expert users to represent their needs according to their expertise and specific organisational context. This section shows a case study where security requirements are represented at different levels of granularity. In this case study we only considered qualitative weights to indicate the Customer's relative priorities (*High-Important*, *Low-Important* and *Medium-Important*) corresponding to the numeric values 1, 0 and 0.5 respectively.

We also assume a Customer denoting controls *Audit Planning*, *Independent Audits* and *Third Party Audits* as *High-Important*, *Low-Important* and *Medium-Important* respectively. *High-Important* for *Facility Security*, and specified low level requirements for *Risk Management* as shown Table 5. Since *Audit Planning* is assigned *HI*, the respective weight is set to 1. On the other hand, *Third Party* is denoted *LI* by the Customer where the respective weight is set to 0. Therefore, PV_{CO1}, PV_{CO2} and PV_{CO3} are aggregated with Customer defined normalized weights (w_{CO}) using Equation (2) such that:

$$w_{CO} = \begin{matrix} CO1 & CO2 & CO3 \\ \begin{pmatrix} 0.67 & 0 & 0.33 \end{pmatrix} \end{matrix}.$$

Therefore, PV_{CO} is:

$PV_{CO} = (\,0.2615 \quad 0.2154 \quad 0.2615 \quad 0.2615\,)$.

This implies that CSP_2 does not fulfill CSU Compliance SLO and both CSP_1 and CSP_2 equally satisfy that requirement. For *FS*, the user specified *High-Important* which is assigned as 1 for all security SLOs.

$$PV_{FS} = \begin{matrix} CSP_1 & CSP_2 & CSP_3 & CSU \\ \begin{pmatrix} 0.2121 & 0.1970 & 0.29545 & 0.29545 \end{pmatrix} \end{matrix}.$$

Similarly, as Case I *Risk Management* is evaluated such that:

$$PV_{RI} = \begin{matrix} CSP_1 & CSP_2 & CSP_3 & CSU \\ \begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix} \end{matrix}.$$

Subsequently, PV_{CO}, PV_{FS} and PV_{RI} are aggregated to obtain the total secSLA priority vector:

$PV_{total} = (\,0.2412 \quad 0.2208 \quad 0.2690 \quad 0.2690\,)$.

Therefore, only CSP_3 satisfies the Customer needs while both CSP_1 and CSP_2 do not fulfill Customer requirements, as shown in Fig. 6. That was expected, as CSP_1 is not providing *FS1.1* and CSP_2 is under-provisioning *CO1.2* and not providing FS1.2.

5.1.3 Case III: A Non-Expert Customer

In this case study, the Customer represents his security requirements at a coarse-grained level (i.e., Control

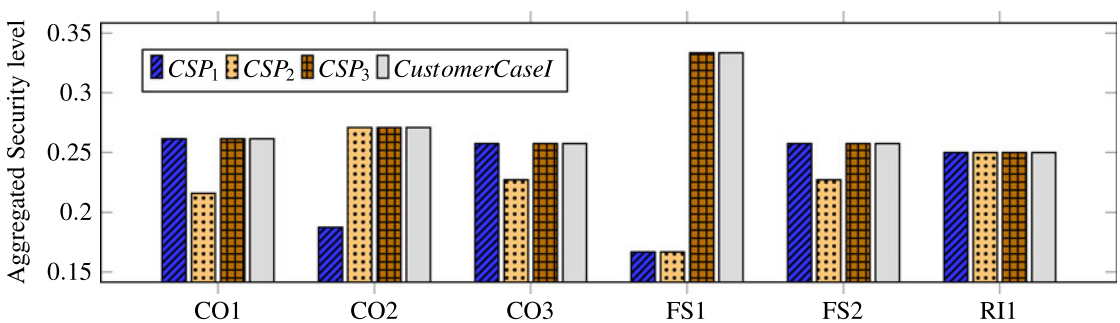


Fig. 9. Using QHP to compare CSP's with respect to customer case I requirements at the SLO level.

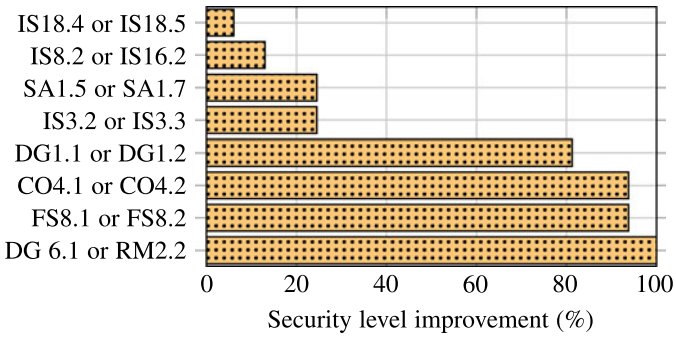


Fig. 10. Sensitivity analysis: CSP_1 SLOs that maximise the overall security level.

Category). For this purpose, the Customer weights *High-Important* for *Compliance*, *Low-Important* for *Facility Security* and *Medium-Important* for *Risk Management* at the Control Category level. Similarly, as shown in previous cases, the priority vectors of *CO*, *FS*, and *RI* are aggregated with Customer normalized defined weights (w_{total}) using Equation (2):

$W_{total} = (\,0.67 \quad 0 \quad 0.33\,)$ where:
$PV_{CO} = (\,0.2254 \quad 0.2279 \quad 0.2734 \quad 0.2734\,)$
$PV_{RI} = (\,0.2250 \quad 0.2250 \quad 0.2750 \quad 0.2750\,).$

Therefore, the total priority vector is:

$PV_{total} = (\,0.2253 \quad 0.2269 \quad 0.2813 \quad 0.2813\,).$

As in the previous cases, only CSP_3 satisfies the Customer needs. However, as observed, the CSPs ranking was different than from previous cases. In this case, CSP_2 outperforms CSP_1. This result was expected as the Customer assigned weights only at the Category-level and Facility Security is assigned *Low-Important*, which affected the overall evaluation. Moreover, CSP_2 is under-provisioning *CO1.2* and CSP_1 is not providing *CO2.1*.

5.2 The CSP Perspective: Maximising Offered Security Levels

The second validation scenario presented in this section applies the secSLA evaluation techniques to solve problems faced by CSPs i.e., (a) which specific security SLOs from the offered secSLA should be improved in order to maximise the overall security level?, and (b) how to improve their service security level to meet the Customers requirements? This might be the case of a well-established CSP deciding where to invest in order to achieve the highest possible security level, or a new CSP designing the secSLA. To answer these questions, we performed two sensitivity analyses to ascertain the security benefits of improving one or more SLOs. The presented sensitivity analysis can be performed using QPT or QHP, however this section applies only QHP given the flexibility it offers for evaluating secSLAs at different levels of granularity and its suitability for implementing what-if scenarios (cf., Section 4.3).

The experiments used the CSP_1 dataset described at the beginning of this section (139 SLOs based on CSA STAR), and applied the Case I requirements to setup the Customer's baseline for the security evaluation. From the existing 139 SLOs the CSP1 is under-provisioning 80 of them. Fig. 10 shows how the QHP technique can be used to analyse an

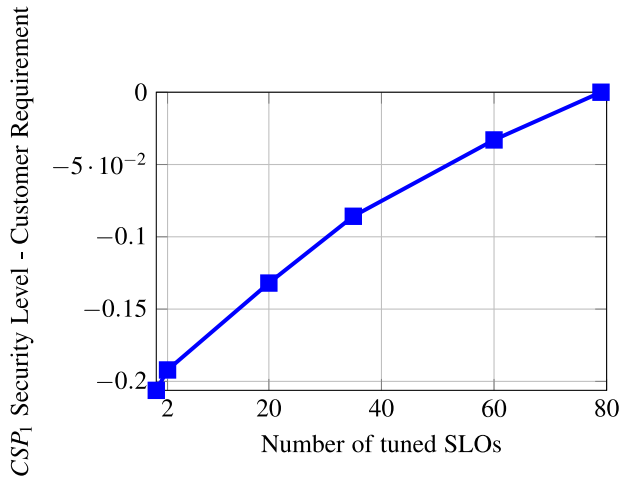


Fig. 11. Sensitivity analysis: combined security effect of sets of SLOs.

existing secSLA, and extract the individual SLOs that if enhanced would result on different improvements associated to the overall security level. In this case, the X-axis represents the improvement associated to the overall security level after enhancing any of the SLOs. It is shown as a percentage where 0 percent corresponds to the original secSLA and 100 percent is the most effective SLO. For example, providing tenants with the security policies applicable to virtualised resources (RM2.2 in Fig. 10), quantitatively increases CSP_1 security level better than improving the thresholds committed for any of the encryption-related SLOs IS18.4 or IS18.5. Also as observed in the figure, improving the SLO DG6.1 would result exactly in the same quantitative improvement than RM2.2's. In this case, the CSP might need to use additional criteria (e.g., economic cost associated with the proposed changes to the secSLA) in order to take a decision related to the SLO to enhance.

The second sensitivity analysis considers the combined security effect of improving simultaneously two or more of the SLOs under-provisioned by CSP_1, based on the Customer requirements of the Case I. Results of the analysis are shown in Fig. 11, where it can be observed how the security level of the CSP approaches faster to the Customer requirement (i.e., $Yaxis = 0$) if several of its offered SLOs are enhanced at the same time. Of course, if all 80 under-

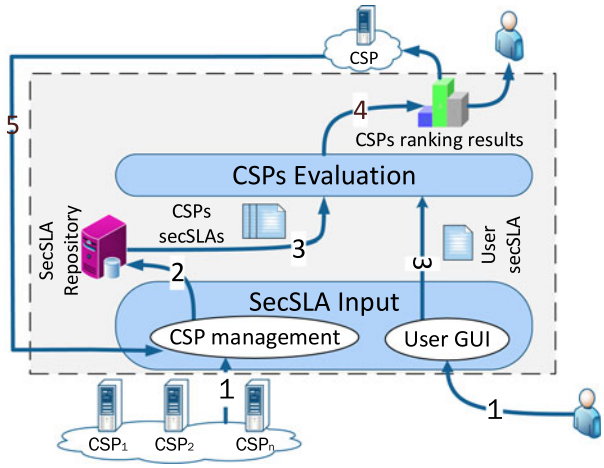
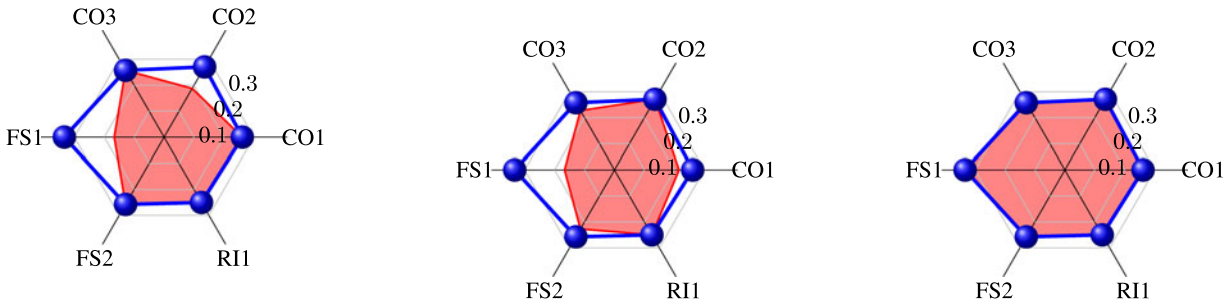


Fig. 12. Architecture of the secSLA dashboard.



(a) CSP_1 wrt $Customer$ (b) CSP_2 wrt $Customer$ (c) CSP_3 wrt $Customer$

Fig. 13. CSPs comparison with respect to the customer case I requirements for different security controls.

provisioned SLOs are improved then the security level of CSP_1 exactly matches the Customer requirement.

6 Decision-Making secSLA Dashboard

As part of our research on Cloud security secSLAs, we developed the dashboard that implements the QHP technique presented in Section 4. This "secSLA dashboard" allows prospective Cloud Customers to define their security requirements in order to graphically compare different CSPs based on their offered secSLA. The dashboard also implements a graphical interface that can be used by CSPs to add their secSLA into a trusted repository.

At the core of our secSLA dashboard are the following building blocks (please also refer to Fig. 12):

1) Customer GUI: Customers are allowed to specify their requirements and assign their priorities at varied levels of the hierarchical representation in order to obtain the required secSLAs. Both requirements and priorities are entered by customers in compliance with the description presented in Section 5.
2) CSP GUI: Once the CSP has uploaded its information[10] to CSA STAR it can be retrieved via the *Download Manager*. Afterwards, this report is used to manually create the corresponding CSP secSLA and store it into a trusted *secSLA Repository* via the *secSLA Management* module (Step 2). This module is also used to update, delete and modify stored Cloud secSLAs.
3) secSLA Repository: This database stores the secSLAs used by the dashboard in easy to use XML format. Also as future work, we will integrate a set of "protocol adapters" to automatically insert/retrieve data from the secSLA Repository from multiple sources e.g., an HTTP adaptor to download the secSLA from the CSP website.
4) Security Evaluator: this module retrieves, from the *secSLA Repository*, the Cloud secSLAs to benchmark with respect to the user-defined security requirement. Based on the customer preferences, the evaluation can take place with either QPT or QHP, depending on the required functionality (cf., Section 4.3) (Step 3). The obtained results are visualized (bar and radar charts) via the *Dashboard* (Step 4). CSPs have the possibility to analyse their secSLAs (to better fulfil the Customers requirements) and update their secSLAs stored in the repository using the CSP management module (Step 5).

The Dashboard allows Customers to compare CSP's based on a set of security requirements, although also CSP's can benefit from comparing their secSLA with respect to others providers. The obtained results are shown using bar charts (as shown in Figs. 6, 7, and 9) and spider charts as shown in Fig. 13 (where from a coarse-grain perspective a Customer can observe that CSP_3 fulfils her security requirements better than CSP_1 and CSP_2). The actual dashboard will be publicly available post-publication.[11]

7 Conclusion

This paper has extended two state of the art security evaluation techniques (namely QPT and QHP) to quantitatively assess the security level provided by Cloud secSLAs. The proposed extensions were designed based on the specifics of secSLAs as defined by state of the art works and standardisation bodies. Furthermore, both QPT and QHP were empirically validated through a couple of case studies using real-world CSP data obtained from the Cloud Security Alliance. The validation experiments were useful to highlight the advantages and limitations of these techniques, and provided an objective comparison of both QPT and QHP in order to guide (prospective) adopters. This paper also presented the prototype of a decision-making security dashboard that implements the discussed evaluation techniques, to allow customers visually comparing CSPs based on their offered secSLAs.

As future work, we plan extensions to QPT and QHP in order to implement advanced security metrics/Cloud secSLA notions e.g., uncertainty, end-to-end security evaluation (CSP composition), and dependencies within secSLAs elements (e.g., controls, SLOs). The lack of real-world information (including standards and best practices) needed to empirically validate these advanced notions will become an important challenge to overcome e.g., through the CSP community of the Cloud Security Alliance.

Because secSLAs are concrete mechanisms to improve security assurance and transparency in Cloud systems, our belief is that their quantitative assessment will provide a critical element to drive the development of tools aimed to

10. Also referred as CAIQ reports by the Cloud Security Alliance.

11. Further details related to the operation of the Dashboard can be seen on a video demonstrating the use of the prototype presented in this section on the following link *https://www.youtube.com/watch?v=dU9HiJMC96M*.

empower customers during the whole Cloud service lifecycle (from procurement to termination). From a CSP perspective, techniques like QPT and QHP trigger on the one hand the adoption of advanced secSLA capabilities (e.g., automation and continuos monitoring), and on the other hand compliance with relevant standards in this field.

Acknowledgments

This research was supported in part by EC FP7 SPECS and H2020 ESCUDO-CLOUD.

References

[1] "Guidelines on information security controls for the use of cloud computing services based on ISOIEC 27002," International Organization for Standardization, ISOIEC 27002, 2014.
[2] Cloud Security Alliance. Cloud controls matrix v3. [Online]. Available: https://cloudsecurityalliance.org/research/ccm/, 2015.
[3] "Security and privacy controls for federal information systems and organizations," National Institute of Standards and Technology, NIST 800-53v4, 2014.
[4] R. Sandhu, "Good-enough security: Toward a pragmatic business-driven discipline," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 66–68, Jan. 2003.
[5] "Survey and analysis of security parameters in Cloud SLAs across the European public sector," European Network and Information Security Agency, 2011-12-19, 2014.
[6] "Information Technology-cloud computing? Service level agreement (SLA) framework and terminology (Draft)," International Organization for Standardization, ISO/IEC 19086, 2014.
[7] "Cloud service level agreement standardisation guidelines," European Commission, C-SIG SLA 2014, 2014.
[8] J. Luna, R. Langenberg, and N. Suri, "Benchmarking cloud security level agreements using quantitative policy trees," in *Proc. ACM Cloud Comput. Security Workshop*, 2012, pp. 103–112.
[9] A. Taha, R. Trapero, J. Luna, and N. Suri, "AHP-based quantitative approach for assessing and comparing cloud security," in *Proc. IEEE Conf. Trust, Security Privacy Comput. Commun.*, 2014, pp. 284–291.
[10] A. Li, X. Yang, S. Kandula, and M. Zhang, "Cloudcmp: Comparing public cloud providers," in *Proc. Conf. Appl., Technol., Archit., Protocols Comput. Commun.*, 2010, pp. 1–14.
[11] S. K. Garg, S. Versteeg, and R. Buyya, "A framework for comparing and ranking cloud services," *J. Future Generation Comput. Syst.*, vol. 29, no. 4, pp. 1012–1023, 2013.
[12] J. Siegel and J. Perdue, "Cloud services measures for global use: The service measurement index," in *Proc. Annu. SRII Global. Conf.*, 2012, pp. 411–415.
[13] R. Henning, "Security SLAs: Quantifiable security for the enterprise?" in *Proc. ACM Workshop New Security Paradigms*, 1999, pp. 54–60.
[14] C. Irvine and T. Levin, "Quality of security service," in *Proc. ACM Workshop New Security Paradigms*, 2001, pp. 91–99.
[15] S. Lindskog, *Modeling and Tuning Security from a Quality of Service Perspective*. Gothenburg, Sweden: Chalmers Univ. Technol., 2005.
[16] V. Casola, A. Mazzeo, N. Mazzocca, and M. Rak, "A SLA evaluation methodology in service oriented architectures," in *Proc. Conf. Quality Protection*, 2006, vol. 23, pp. 119–130.
[17] S. A. de Chaves, C. B. Westphall, and F. R. Lamin, "SLA perspective in security management for cloud computing," in *Proc. IEEE Conf. Netw. Services*, 2010, pp. 212–217.
[18] G. Frankova and A. Yautsiukhin, "Service and protection level agreements for business processes," in *Proc. IEEE Workshop Service Oriented Comput.*, 2007, pp. 38–43.
[19] L. Krautsevich, F. Martinelli, and A. Yautsiukhin, "A general method for assessment of security in complex services," in *Proc. 4th Conf. Service-Based Internet*, 2011, pp. 153–164.
[20] K. Bernsmed, M. G. Jaatun, P. H. Meland, and A. Undheim, "Security SLAs for federated cloud services," in *Proc. 6th IEEE Conf. Availability, Rel. Security*, 2011, pp. 202–209.
[21] M. Almorsy, J. Grundy, and A. Ibrahim, "Collaboration-based cloud computing security management framework," in *Proc. IEEE Int. Conf. Cloud Comput.*, 2011, pp. 364–371.
[22] V. Casola, R. Preziosi, M. Rak, and L. Troiano, "A reference model for security level evaluation: Policy and fuzzy techniques," *J. Universal Comput. Sci.*, vol. 11, no. 1, pp. 150–174, 2005.
[23] "(Draft) information Technology-cloud Computing-Service level agreement (SLA) framework and terminology," International Organization for Standardization, ISO/IEC 19086, 2014.
[24] "Cloud standards coordination final report," European Telecommunications and Standards Institute, Tech. Rep., Editor Emmanuel Darmois, 2013.
[25] "The CloudTM Project in cloud computing service level agreements - exploitation of research results," European Commission, Tech. Rep., Editor Dimosthenis Kyriazis, 2014.
[26] "(Draft) cloud computing: Cloud service metrics description," NIST, Tech. Rep., Editor Frederic DeVauix, 2014.
[27] W. Jansen, "Directions in security metrics research," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. TR-7564, 2010.
[28] T. Saaty, "How to make a decision: The analytic hierarchy process," *Eur. J. Operational Res.*, vol. 48, pp. 9–26, 1990.
[29] M. Zeleny, *Multiple Criteria Decision Making*. New York, NY, USA: McGraw Hill, 1982.
[30] Cloud Security Alliance. (2011). The security, trust & Assurance registry (STAR). [Online]. Available: https://cloudsecurityalliance.org/star/

Jesus Luna received the PhD degree from the "Technical University of Catalonia." He is the research director at Cloud Security Alliance (Europe). He is also affiliated with TU Darmstadt with his main research interests as security quantification, cloud security, and security policies.

Ahmed Taha is currently working toward the PhD degree at the Department of Computer Science, TU Darmstadt, Germany. His research interests include cloud security metrics, security assessment, and quantification.

Ruben Trapero received the PhD degree from the Universidad Politecnica of Madrid and was an assistant professor at Universidad Carlos III of Madrid. Since 2014, he is a lead researcher at TU Darmstadt, Germany. His research interests include privacy, identity management, cloud security, and service engineering.

Neeraj Suri received the PhD degree from the UMass-Amherst and is a chair professor at TU Darmstadt, Germany. His research addresses the design, analysis and assessment of trustworthy Cloud services.