Measuring Web Service Security in the Era of Internet of Things

Bo Zhou[a,*], Quan Zhang[b], Qi Shi[a], Qiang Yang[c], Po Yang[a], Yinyan Yu[d]

[a] *Department of Computer Science, Liverpool John Moores University, Liverpool, UK*
[b] *School of Information Engineering, ShenYang University of Technology, Shenyang, China*
[c] *College of Electrical Engineering, Zhejiang University, Hangzhou, China*
[d] *Institute of Computer Science and Technology, Peking University, Beijing, China*

Abstract

Technologies such as Internet of Things allow small devices to offer web-based services in an open and dynamic networking environments on a massive scale. End users or service consumers face a hard decision over which service to choose among the available ones, as security holds a key in the decision making process. In this paper a base linguistic evaluation set is designed, based on which all the other fuzzy term sets that used for describing security attributes are uniformed and integrated for calculating an overall security value of the services. This work, to the best of our knowledge, is the first practical solution to offer direct comparisons and rankings of network services based on multiple security attributes such as confidentiality, availability, privacy and accountability. We analysed four major cloud service platforms to illustrate the proposed approach.

Keywords: Network Service, Security measurement and evaluation, Quantitative service security, Linguistic evaluation

1. Introduction

In the digital world, a service is defined as a software unit that provides certain functionalities. A web service is a service that is made remotely available

*Corresponding author at: Department of Computer Science, Liverpool John Moores University, Liverpool, United Kingdom.
Email address: b.zhou@ljmu.ac.uk (Bo Zhou)

to other entities through networks. By using standard communication protocols and languages, web services provide necessary interfaces so that any system can invoke them remotely. The Service-Oriented Architecture (SOA) provides designs and frameworks to offer services as self-contained units [1]. One can invoke a web service, as long as the input satisfies the interface specification, and let the output of the web service to be an input to another service if they need to work together. The most commonly used communication protocol for exchanging information between web services is the Simple Object Access Protocol (SOAP) [2]. SOA platforms provide a foundation for modeling new applications, which involves planning, searching for, connecting, and invoking web services.

One of the issues faced by a service consumer is to measure and choose a right service from potentially a very large service pool. Services provided by different providers may offer the same functionality, but they could be very different in terms of cost, quality, or security. Therefore the service consumer faces the dilemma of picking up the most suitable services for his/her application, especially in the era of Internet of Things (IoT) where small devices are made available as service unit through an open and dynamic networking environment in massive scale. Among all the existing works that have been used to quantify and compare web services, we found that most of them focus on the QoS only and the key question of quantifying services based on their security properties remains unanswered. Nonetheless, it is crucial to measure the services from security perspective since one service developed with good faith in its security may not be necessarily good enough for another to use.

In this paper we propose a novel quantitative approach based on fuzzy terms. In particular, we focus on security as it is a big challenge for utilising web services, due to the lack of a common ground and evaluation criteria. It is to use a linguist evaluation method to quantitatively measure services based on their security attributes such as *confidentiality*, *availability*, *privacy* and *accountability*. These attributes are formulized into one base linguistic evaluation set and calculated towards an overall security value. In this way the comparison of different services' security becomes possible.

To the best of our knowledge, this work is the first practical approach to target the issue of evaluating web services based on multiple security attributes at the same time. It provides the foundation for further research into this area and has great potential to be extended to solve similar issues faced by other information systems. The calculation is based on information that already exist, e. g., descriptions in the Service Level Agreements (SLAs) [3], thus it is feasible and practical enough to make immediate impact.

The rest of the paper is organised as follows. The next section explains how web service security is presented in current network and the challenges service consumers are facing. section 3 introduces the linguistic evaluation foundation and the triangular membership function. The next section explains our approach to formulize different linguistic term sets and calculate an overall security value for web services based on multiple attributes. An example is given in section 5 to illustrate the solution and section 6 discusses related work. Finally the paper concludes with an outline of future work in section 7.

2. Web Service Security

2.1. Security wtih SLAs and WSDL

Web services are normally made available together with a Service Level Agreements (SLA). A SLA is a formal guarantee that has to be accepted by service consumers before the service being used. A SLA can specify the properties of a service across different levels. For example, on business level it can describe what kind of functionality the service is offering and how the users will be charged (cost); on technical level it may describe the number of shutdowns the service might experience each year (QoS).

Security can also be promised as part of the SLA. However its coverage is rather poor to date due to the lack of well defined semantics. The SLAs traditionally focus on the QoS metrics such as a bandwidth guarantee and backup strategy. Even when the security being mentioned, in practice it tends to be written in a natural language with fuzzy terms such as “High”, “Good”, etc.

Therefore it is very difficult for the service consumer to really understand the situation and compare the web services from the security aspect.

Apart from SLA, a web service also describes its interfaces through a Web Service Description Language (WSDL). A WSDL file specifies how to invoke the service, i.e. the input parameters, in order to communicate with the service and the expected output for each of the operations provided by the service. The WSDL file can be generated automatically from the web service code. Based on WSDL specification files, a service consumer can design his/her applications accordingly and use SOAP to call the operations listed in the WSDL files.

Although WSDL is mostly used to specify the functional aspects of a service, it is possible to attach non-functional properties such as security to the WSDL. WS-SecurityPolicy [4] is an extension of WSDL to secure SOAP messages. It utilises standards like SAML [5], XML Signature [6], and XML Encryption [7] to achieve the goal of secure communications with web services. WS-SecurityPolicy is different from the Secure Socket Layer (SSL) protocol as the WS-SecurityPolicy only encrypts the content of a SOAP message while SSL can encrypt the entire communication channel. Comparing to SSL, WS-SecurityPolicy is more flexible as it can choose which part of the SOAP message to be encrypted by using which cryptographic algorithm. WS-SecurityPoliy is attached to the WSDL by declaring it in the WSDL.

2.2. Challenges

Despite some efforts from SLA and WSDL, security issue remains a big challenge for web services. The dilemmas faced by a service consumer are in three folds.

- Firstly, security is a broad concept that includes many aspects such as confidentiality and privacy. One service may be stronger than another in terms of confidentiality; while it is also possible that the very same service has weaker protection of privacy. It is a typical multi-criteria issue, which service consumers are not always in the position to resolve due to the lack of expertise.

- Secondly, WS-SecurityPolicy was proposed to secure the SOAP messages. It is well equipped for - but also limited to - the security of communication with web services. Security requirements at higher levels are hard to be expressed by using the WS-SecuirtyPolicy. In contrast, security descriptions in SLAs are more open and inclusive but not always precise, especially in natural language. The situation can get even more complicated when more than one SLA language is involved.

- Finally, although some security modelling and verification techniques allow the service consumer to specify certain security properties that the service has to comply with before the service being used [8], in practice the number of services that satisfy the security requirements could still be very large. Therefore the service consumer still needs help to be able to make a sensible choice from a potentially very large pool of services.

In this paper, we try to solve the problem from the angle of linguistic evaluation, i. e., measuring the fuzzy linguistic terms used in the SLAs in a more mathematical and scientific way. We use the most common security attributes as example, so the method can be applied to existing services requiring little changes.

3. Linguistic Evaluation Foundation

3.1. Problem Description

In this section, we first explain how to mathematically describe the problem of evaluating a SLA in terms of security. Before comparing two services, i. e., their SLAs in our study, they must be formulated first. In order to facilitate the issue, the following symbols are used to describe the problem. Here we try to evaluate the security of web services with multiple attributes described in their SLAs.

$S = (S_1, S_2, ..., S_m)$ represents a set of alternative web services, where $m \geq 2$.

$C = (C_1, C_2, ..., C_n)$ represents a set of the security attributes, e. g., *confidentiality*, *availability*, *privacy*, and *accountability*, where $n \geq 2$.

$W = (w_1, w_2, ..., w_n)$ represents a weight vector of the attributes, where w_j is the weight of attribute C_j, $0 \leq w_j \leq 1$ and $\sum_{j=1}^{n} w_j = 1$.

$\tilde{A} = [\tilde{a}_{ij}]_{m \times n}$ is the decision matrix with linguistic attribute values, where $\tilde{a}_{ij}$ denotes the linguistic evaluation on alternative S_i against attribute C_j, where $i = 1, ..., m$, and $j = 1, ..., n$. Since attributes have different characteristics, the linguistic evaluation sets used for describing them are not the same. For example, we use terms from "low" to "high" to evaluate *confidentiality* and use different set of terms from "weak" to "strong" to measure *privacy*. Therefore, in the linguistic decision matrix $\tilde{A} = [\tilde{a}_{ij}]_{m \times n}$, the evaluations on the services against different security attributes may come from different linguistic sets, and they are also of different granularities. Thus, the linguistic attribute values have to be uniformed in order to make the comparisons possible. This is described in section 4. To simplify the issue, in this paper we assume all the alternative services are described using the same SLA language. In practice the situation may get further complicated when more than one language being used.

3.2. Concepts of Linguistic Evaluations

3.2.1. Value of Fuzzy Terms

In a complex or uncertain decision environment, fuzzy languages can be used to express decision makers' subjective opinions or judgments more precisely [9, 10]. Security is one of the subjects which people may interpret differently, depending on their knowledge levels and experiences. For example, when describing the *confidentiality* level of a service, the terms like "low", "fair" and "high" can be used in the SLA and in practice, the service consumers are likely to accept these fuzzy terms even in business cases due to the lack of precise understanding and definition of these attributes and terminology. It is fine when there is only one single attribute to be considered, as the comparison is straightforward, e. g., a service with the "high" confidentiality level is certainly preferred over another service with the "low" confidentiality level. The problem arises when more than one security attribute being taken into account, which is often the case in real world. This then becomes a multi-criteria decision making

issue and the fuzzy terms must be mapped to real numbers first in order to allow a fast and accurate comparison. To measure the real level of a particular term in a fuzzy language set, a triangular membership function is commonly used for the mapping [11]. Similar to our previous work in paper [12], we use the following definition and equation to represent the membership function.

Definition 1. *A linguistic term $\tilde{T}$ on a real number set is defined as a triangular fuzzy number denoted as (u, α, β), and its membership function $\mu_{\tilde{T}}$ is defined as:*

$$\mu_{\tilde{T}}(x) = \begin{cases} \dfrac{x-\alpha}{u-\alpha}, & x \in [\alpha, u], \\ \dfrac{x-\beta}{u-\beta}, & x \in [u, \beta], \\ 0, & otherwise, \end{cases} \tag{1}$$

where $\alpha < u < \beta$, u is the model value, α and β stand for the lower and upper values of linguistic term $\tilde{T}$ respectively. The triangular membership function is shown in Figure 1. In our study, the values of u, α, and β are determined by the size of the linguistic term set, which we will explain in the next sub-section.

3.2.2. Linguistic Term Set

When alternative services are evaluated against the security attributes, the linguistic term sets for the security attributes should be determined first. As we already mentioned, different linguistic term sets are employed for describing different attributes. In this study we assume the SLA uses predefined schema and ontology to constrain the space of the term sets.

Suppose $TERMSET = (t_0, t_1, ..., t_g)$ is a linguistic term set for evaluating one attribute of the services. The $TERMSET$ is defined as an ordered set, which is composed of *g+1* linguistic terms. For example, consider a set of five terms $TERMSET = (t_0 =$ "*none*"$, t_1 =$ "*poor*"$, t_2 =$ "*average*"$, t_3 =$ "*good*"$, t_4 =$ "*excellent*"), the membership functions of this term set is drawn in Figure 2. It assumes the five terms on real numbers are equally distributed over the range from 0 to 1. Taking term "average" as an example, the values of its membership

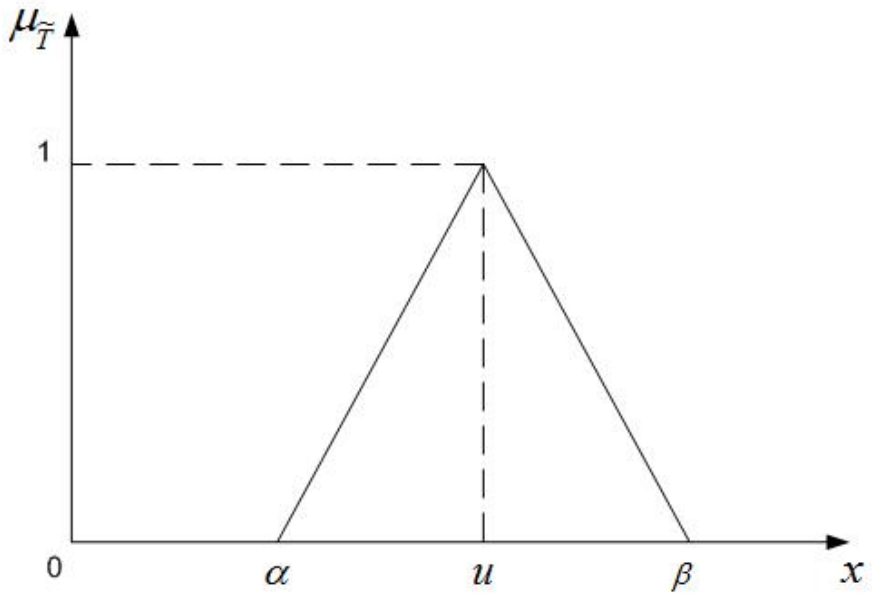

Figure 1: Triangular membership function.

function (u, α, β) are defined as (0.5, 0.25, 0.75). Given $x = 0.5$, apply these values to Equation 1 to get $\mu_{\tilde{T}}(x) = 1$ on term "average".

In addition, the following properties of the *TERMSET* are also assumed in the membership function study, similar to papers [13] and [14]:

- Firstly, the *TERMSET* is ordered:

  $$t_i \geq t_j, \text{ if } i \geq j$$

  where symbol "$\geq$" denotes "better or equal".

- Secondly, there is a negation operator "Neg":

  $$Neg(t_i) = t_j, \text{ if } j = g - i$$

  where $g + 1$ is the number of elements in the *TERMSET*, and the largest term in *TERMSET* is t_g.

- Thirdly, there is a "Max" operator and a "Min" operator respectively:
  $Max(t_i, t_j) = t_i$ and $Min(t_i, t_j) = t_j$, if $t_i \geq t_j$

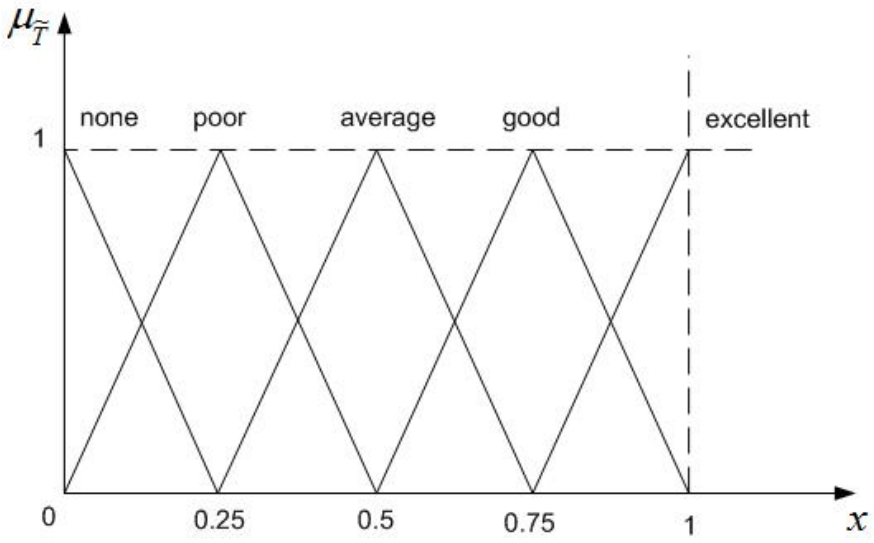

Figure 2: Membership function of TERMSET with five terms.

3.2.3. Base Linguistic Evaluation Set

As we mentioned earlier, in the linguistic decision matrix $\tilde{A} = [\tilde{a}_{ij}]_{m\times n}$ different linguistic term sets with various granularities are applied to different security attributes to suit their characteristics. Therefore these linguistic term sets are not comparable and have to be uniformed. In this study, a special term set of seven terms {"lowest", "lower", "low", "fair", "high", "higher", "highest"} is adopted as a base linguistic evaluation set, to evaluate the security level of the services. Different attribute values are uniformed into the base evaluation set (we will explain the details in section 4). In order to facilitate descriptions, $TERMSET^B = \{term_0^B, term_1^B, ..., term_g^B\}$ is used to denote the base linguistic evaluation set.

With $TERMSET^B$, the services' security is distinguished more detailed and meticulously. The most insecure services are classified as class "lowest". If a specific classification (e. g., with more elements) is needed, it can be determined based on specific problems, which do not affect the viability of the proposed approach in the next section.

4. Quantify and Rank the Web Services

4.1. Uniform different linguistic sets

In this subsection we explain how to uniform different linguistic sets with various granularities into the base linguistic evaluation set $TERMSET^B$.

Suppose $TERMSET^j$ ($g+1$ is the cardinality) is the linguistic evaluation set corresponding to attribute C_j. For the linguistic evaluation value $term_i^j$ of service S_i with $term_i^j \in TERMSET^j$, $i = 1, ..., m$, and $j = 1, ..., n$, the following function τ can be used to transform $term_i^j$ into the fuzzy set over $TERMSET^B$ [15]:

$$\tau : term_i^j \rightarrow F_i^j(TERMSET^B) \tag{2}$$

where $F_i^j(TERMSET^B)$ is the fuzzy set over $TERMSET^B$, and,

$$\begin{aligned} \tau(term_i^j) = \{&(term_0^B, \gamma_0^{ij}), (term_1^B, \gamma_1^{ij}), ..., \\ &(term_g^B, \gamma_g^{ij})\} \end{aligned} \tag{3}$$

γ_l^{ij} is the shared maximum value of membership functions of $term_i^j$ and $term_l^B$. In mathematics γ_l^{ij} is expressed as:

$$\gamma_l^{ij} = \underset{x}{Max Min}\{\mu(term_i^j), \mu_l(term_l^B)\}, l = 0, 1, ..., g \tag{4}$$

where $\mu(term_i^j)$ and $\mu_l(term_l^B)$ are the membership functions of term $term_i^j$ and $term_l^B$ respectively. $x \in [0, 1]$ is a real number shared by $term_i^j$ and $term_l^B$ in their triangular membership functions.

To explain the meaning of these equations, we use the same example in Figure 2 that has a term set of five values {"none", "poor", "average", "good","excellent"}. Assume a service (S_2) has the attribute (C_3) with value of "poor", and the base linguistic evaluation set has seven terms {"lowest", "lower", "low", "fair", "high", "higher", "highest"}. According to Equation 3, the uniformed result of attribute C_3 of service S_2 is expressed as:

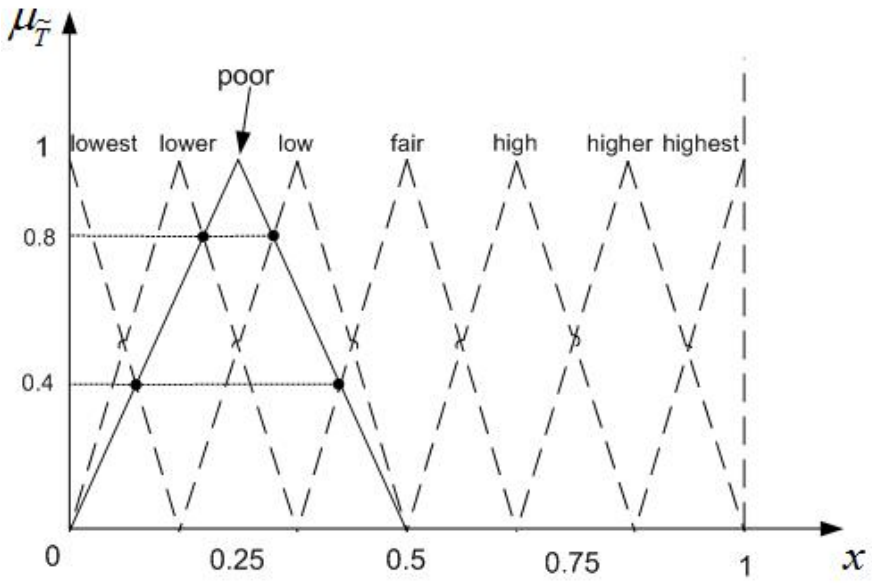

Figure 3: Uniform the value "poor" in one term set to the base linguistic evaluation set.

$$\tau(term_2^3) = \{(lowest, \gamma_0^{23}), (lower, \gamma_1^{23}), (low, \gamma_2^{23}),$$
$$(fair, \gamma_3^{23}), (high, \gamma_4^{23}), (higher, \gamma_5^{23}), (highest, \gamma_6^{23})\}$$

where the values of $\gamma_l^{23}(l = 0, ..., 6)$ can be calculated based on Equation 4, or with the help of Figure 3. It is literally to find the highest crossover point of the "poor" membership function, with other membership functions in the base linguistic evaluation set. In this example, the "poor" membership function gets crossed with four values ("lowest", "lower", "low", and "fair") in the base set. Therefore the final expression of $\tau(term_2^3)$ is:

$$\tau(term_2^3) = \{(lowest, 0.4), (lower, 0.8), (low, 0.8),$$
$$(fair, 0.4), (high, 0), (higher, 0), (highest, 0)\}$$

It reflects the position of value "poor" in its original term set, i.e., since

the service S_2 has the attribute C_3 with value "poor", its uniformed result is mapped mainly into the lower part of the base linguistic evaluation set.

In this way, in the linguistic decision matrix $\tilde{A} = [\tilde{a}_{ij}]_{m\times n}$ with multiple granularities, the attribute values $\tilde{a}_{ij}$ of $term_i^j$ from different linguistic evaluation sets, are all uniformed into the fuzzy set over $TERMSET^B$ denoted as $F_i^j(TERMSET^B)$.

4.2. Calculate the Overall Security Value of Web Service

Based on the above discussion, after the linguistic decision matrix $\tilde{A} = [\tilde{a}_{ij}]_{m\times n}$ is uniformed, the weighted sum method can be used to calculate the overall security values of the alternative services:

$$Overall_i = \sum_{j=1}^{n} (F_i^j(TERMSET^B) \times w_j), i = 1, ..., m \tag{5}$$

where, w_j is the weight of attribute C_j and $j = 1, ..., n$. The weight vector of attributes can be given by decision makers based on experience or determined by the AHP method [16].

For example, assume a service is described by four security attributes - *confidentiality* (term_1), *availability* (term_2), *privacy* (term_3), and *accountability* (term_4). Their values are uniformed into the base evaluation set respectively as the following:

$$
\begin{aligned}
\tau(term_1) =&\{(lowest, 0.4), (lower, 0.8), (low, 0.8),\\
&(fair, 0.4), (high, 0), (higher, 0), (highest, 0)\}\\
\tau(term_2) =&\{(lowest, 1), (lower, 0.67), (low, 0.33),\\
&(fair, 0), (high, 0), (higher, 0), (highest, 0)\}\\
\tau(term_3) =&\{(lowest, 0), (lower, 0), (low, 0),\\
&(fair, 0), (high, 0), (higher, 0.6), (highest, 1)\}\\
\tau(term_4) =&\{(lowest, 0), (lower, 0), (low, 0.6),\\
&(fair, 1), (high, 0.6), (higher, 0), (highest, 0)\}
\end{aligned}
$$

Assume the weight set W of these four attributes is defined as $W = (0.1, 0.5, 0.3, 0.1)$ by the service consumer. We calculate the *Overall* security value of the service over the base evaluation set based on Equation 5. For example, on term "lowest", the overall value is:

$$0.4 \times 0.1 + 1 \times 0.5 + 0 \times 0.3 + 0 \times 0.1 = 0.54$$

Similarly we can get the *Overall* value over the base evaluation set as:

$$
\begin{aligned}
Overall = \{&(lowest, 0.54), (lower, 0.415), (low, 0.305),\\
&(fair, 0.14), (high, 0.06), (higher, 0.18), (highest, 0.3)\}
\end{aligned}
$$

4.3. Rank the Services

As discussed above, the overall value $Overall_i$ of alternative service S_i obtained based on Equation 5 is still a fuzzy set of values over the base linguistic evaluation set $TERMSET^B$. It can be represented as below:

$$
\begin{aligned}
Overall_i = \{&(term_0^B, o_0^{ij}), (term_1^B, o_1^{ij}), ...,\\
&(term_g^B, o_g^{ij})\}, i = 1, ..., m
\end{aligned}
\tag{6}
$$

To compare the alternative services fast and easily, a single-point overall security value d_i of service S_i is needed. In this paper, we use the same method employed by paper [15] and [12] to determine the value d_i.

$$d_i = \frac{\sum_{k=0}^{g} k o_k^i}{\sum_{k=0}^{g} o_k^i}, i = 1, ..., m \tag{7}$$

Continuing the same example from last subsection, the final single-point overall security value for the service is calculated as:

$$\begin{aligned} d &= \frac{0 + 0.415 + 0.61 + 0.42 + 0.24 + 0.9 + 1.8}{0.54 + 0.415 + 0.305 + 0.14 + 0.06 + 0.18 + 0.3} \\ &= 2.26 \end{aligned}$$

In this way, all the alternative services can finally be ranked in descending order based on their values of d_i.

5. ILLUSTRATION

In this section we demonstrate how to use the proposed method to evaluate alternative web services in real world scenario. We look into four major cloud service providers and their security promises - Amazon Web Services [17], Dropbox [18], Google Cloud Platform [19] and Microsoft Azure [20]. It is worth noting that the security features we referred here are publicly available on their websites. However these are not part of the legally bounded SLAs. Instead these features are used as selling points of the cloud services. In some ways, it proves our point that security is not properly covered in the SLAs and the situation has to be changed. Challenging the service providers in this front will actually make the cloud services more transparent to service consumers and improve their security.

We analyse these four cloud services or platforms against four security related attributes, i.e., *Confidentiality* (C_1), *Availability* (C_2), *Privacy* (C_3) and

Accountability (C_4). This is not an exhaust security attribute list, but a rather important and common one when it comes to measure the security of a cloud service. It actually covers a wide range of security features. For example, *confidentiality* covers properties like encryption algorithms employed by the service provider; *availability* evaluates its backup strategy as well as protection solution against DoS attacks; *privacy* indicates the strength of the service's access control mechanism and security compliance; *accountability* measures the facility for post-forensics.

For illustration purposes, we assume the four security attributes use the following term sets in evaluation:

- Confidentiality: {"none", "low", "average", "high", "very high" }
- Availability: {"very poor", "poor", "fair", "good", "very good"}
- Privacy: {"very weak" , "weaker", "weak", "fair", "strong", "stronger", "very strong"}
- Accountability: {"not accountable", "poor", "fair", "good"}

In this paper we evaluate these cloud services simply based on the information they provide and our expertise. Most attributes are mentioned in natural language descriptions and we have to interpret them ourselves. Take the *privacy* issue as an example, the privacy principle by Google states "we work hard to make sure any innovation is balanced with the appropriate level of privacy and security for our users", which reads not very clear and accountable to us; Amazon committed to 'adhere to the Safe Harbor Privacy Principles agreed upon by the U.S., the European Union, and Switzerland"; Dropbox says they comply with the same Safe Harbor program but will also share information with "others working for Dropbox - Dropbox uses certain trusted third parties to help us provide, improve, protect, and promote our Service"; Microsoft also supports the Safe Harbor program, as well as EU Model Clauses and ISO/IEC 27018. Based on the information collected, Google is valued as "weaker" in *privacy* and

Table 1: Evaluation results of Four Major Cloud Services/Plaforms

Services	Attributes			
	Confidentiality	*Availability*	*Privacy*	*Accountability*
Amazon	high	good	strong	good
Dropbox	average	fair	fair	poor
Google	very high	poor	weaker	good
$Microsoft$	low	very good	very strong	fair

Microsoft gets the "very strong" from us. Similarly, in terms of *confidentiality*, Microsoft "offers a wide range of encryption capabilities *up to* AES-256" to store data. The subtle wording of "up to" is enough to see Microsoft get lower confidentiality value comparing to Amazon who allows users to choose AES-256, and Google gets the highest evaluation by promising to apply AES-256 by default. It is also possible to measure a security attribute based on a quantitative value. For example, the service *availability* is specifically mentioned in some SLAs in real number, e. g., 99.98%.

In summary, we analysed the available security descriptions offered by these cloud services. Their security attributes were collected and evaluated with results shown in Table 1. If we compare these services pairwise, it is obvious that Amazon is better equipped in terms of security than Dropbox, as all its attributes are stronger. Apart from this, there is no clear winner as they all have strengths and weaknesses, comparing to its peers. However by apply our aforementioned method, we can get a single-point overall security value for each of the cloud services as $Amazon = 4.65$, $Dropbox = 2.67$, $Google = 3.27$, $Microsoft = 3.83$. Thus, the ranking of these cloud services is $Amazon > Microsoft > Google > Drobox$.

We want to stress that these results are subjective judgements and it will be better if all the security attributes are clearly specified in the SLAs. The four attributes used here may seem rather abstract. This is because there is no agreed ontology on security properties that could potentially appear in the SLAs, and

the information we gathered is quite disparate. Nonetheless, in practice the proposed method can also operate on more concrete security properties such as encryption algorithms directly, as long as these security properties are described by all the alternative services using a finite term set, which satisfies the three assumptions made in section 3.

6. Related work

We see three areas of related work: 1. expressing security attributes in SLAs, 2. ranking web services, and 3. multiple attribute decision making.

6.1. Express Security in SLAs

As we already explained in section 2, SLAs are more flexible in terms of defining complex security requirements. The problem with SLAs is the lack of a common ground for the expression and interpretation of security. Crucially this makes it very difficult to make the SLAs machine readable. Some works have been done in the past in order to express the security features of web services in the SLAs and help the consumers to compare the web services in an automatic way.

Paper [21] was among the first works trying to address the quantifiable security issue in SLAs. Basically it tries to express and measure the security of a service by associating it with performance related metrics. For example, a security requirement of "restore backed up data" is measured by the quantifiable metric of "data restored 95% of time within response time". The way the security has been expressed is rather subjective, depending on the scenario of each enterprise, where the research was targeting. Therefore the process cannot be implemented automatically. Instead, it requires a close study of the enterprises configurations by security specialists.

Paper [22] uses SecAg as another framework to express security metrics in SLAs. SecAg extends the standard WS-Agreement to provide necessary semantics for specifying security properties. For example, with the extensions it can

specify which service level objective (SLO) is auditable and assign an access control list to the SLO. Based on the extensions, the author also proposed a risk-based approach for service matchmaking. Each SLO is assigned a weight w representing the risk that the SLO is not fulfilled. By calculating the weighted Euclidean distance of each SLA to the security requirements using techniques such as a text similarity analyser, the SLA that is closest to the security requirements will be selected as the risk is at the minimum.

For cloud consumers, before employing any cloud service they have to make sure that the service is compliant with their security requirements. In addition, business users seek for assurance that the cloud service they use complies with both industrial standards and government legislations. Unfortunately, SLAs are often not rich enough or directly linked with such legislations or standards, in order to support the compliance check. Paper [23] solves the issue by proposing a compliance vocabulary to embed security controls in the SLAs of cloud services. This vocabulary is associated with the security controls from governance documents. Therefore the SLAs become more transparent to the consumers in terms of the level of security being offered.

6.2. Rank Web Services

After expressing the security in the SLAs, it is still necessary to compare and rank the web services based on the consumer's requirements. In the past the focus was on raking web services based on just their QoS metrics and trying to find the best matched one.

Paper [24] proposed a Web Service Relevancy Function (WsRF) to measure the relevancy ranking of a particular web service based on the user's preferences (weights) and the QoS metrics such as Response Time, Throughput, Availability and Cost. It uses a simple mathematical matrix to normalize the QoS metrics of web services. The method is suitable for QoS metrics that have real numbers. However as security is often described by fuzzy terms, the application of this method is limited. Similarly, paper [25] uses a Singular Value Decomposition (SVD) based technique, and a user assisted weighting system to find higher

order correlations among web services. This enables the selection of web services without an exact match of required QoS attributes.

Paper [26] ranks web services under multi-criteria matching. It targets at accurate web service selection and assigns a dominance score to each advertised web service. Security unfortunately was not the research focus. Other similar works include paper [27], which defines a business-focused ontology to enable semantic matchmaking in open cloud markets.

Paper [28] proposed the concept of Quality of Security Service. It treats the security as part of QoS requirements. The author argues that security requirements such as the strength of a cryptographic algorithm, the length of a cryptographic key, security functions, confidence of policy-enforcement and the robustness of an authentication mechanism would all be specified and measured as the quality of security services. However no explicit example was given in the paper.

Paper [29] proposed an AHP (Analytic Hierarchy Process) based framework for web service quality evaluation. It uses a quality meta-model to format SLAs and assigns weights to different quality characteristics based on their importance. The web services are measured by a satisfaction function, which covers both measurable and non-measurable characteristics. For example, the property of *confidentiality* is measured by combining the encryption algorithm, key length and key protection used. The web service that has the greatest value in the satisfaction function will be chosen. Although not all the security attributes can be calculated in this way, we can benefit a lot from this idea.

Paper [30] proposes a method for finding semantically equal SLA elements from differing SLAs by utilizing several machine learning algorithms. The user requirements are specified in a SLA template, which are compared to different SLAs offered by various service providers. The offered SLAs can be specified in different languages. This method tries to map the elements in different SLAs and generates an equivalence probability score. The cloud service that has the highest score will be selected for the users automatically.

6.3. Multiple Attribute Decision Making (MADM)

The problem this paper tries to address is a MADM issue, where alternatives are always evaluated against some non-commensurate and conflicting attributes. How to rank the alternatives or select the best one has attracted many researches in different disciplines.

Security information on alternatives can be described in different types of formats. Paper [31] transforms evaluation information of alternatives into fuzzy preference relations. Also utility values of alternatives are converted into fuzzy preference relations for ranking as well. After the information from multiple sources are uniformed, fuzzy majority method with fuzzy quantifier are used to aggregate these uniformed evaluation information into a social one and to select the best acceptable alternative. With fuzzy preference relation being the basic format of the decision makers' opinions, [32] propose an approach to calculating the group consensus based on the concept of fuzzy majority. The linguistic quantifiers are employed to represent a fuzzy majority. The group consensus is "soft" based on the fuzzy linguistic quantifiers, which is determined by the decision makers' subjective attitudes.

The security descriptions, in truth, can be expressed in different formats due to different culture and education backgrounds. Therefore uniformity and aggregation process are needed to determine the optimal alternative. Integration is an important task in decision support process, as well as it does in group decision making process with preference information on alternatives. By using the nonparametric Wilcoxon statistical test, paper [33] presents a detailed experimental study on comparing five most widely used distance functions for measuring the consensus in group decision-making problems.

Our work is very much inspired by paper [15], where the author proposed a support system model for reaching the consensus in group decision-making problems where experts express their opinions in linguistic preference relations with multiple granularities. By means of designing the basic linguistic term set, multigranular linguistic information is uniformed.

Finally, with fuzzy preference relation and multiplicative preference relations

as the formats of information sources respectively, [34] proposed two methods for determine the weights of sources, i.e. goal programming model and quadratic programming model. Then two iterative algorithms are developed for group decision making to reach the consensus, respectively.

In summary, to the best of knowledge, there is still no effective solution yet to measure web services from the security perspective. Our solution considers the current situation and uses the widely available SLAs as the foundation to evaluate and compare different web services. We tackle this issue based on a MADM approach and make the web services are directly comparable by analyzing the fuzzy terms used to describe their security in the SLAs. It has been proved to be both practical and effective.

7. Conclusions and Future Work

Internet becomes a world full of web services and IoT devices with networking capability. Through advanced network techniques such as Information-Centric Networking, service consumers are offered wide range of choices for their application. However, measuring and choosing the most appropriate services is not easy, when security of the services is considered in the process.

In this paper we proposed a novel approach to measure and quantify the security attributes of web services based on existing descriptions in the SLAs. In order to extend our work, we need to define a SLA schema that can describe the security attributes in a more consistent and precise way. So that an automatic process can be used to facilitate the process of measuring and comparing large number of web services or networked devices in the era of IoT.

To further prove our ideas we also plan to carry out a real experiment with experts on real decision support tasks. In particular, this evaluation could provide insights into the following two aspects: 1) The process of defining linguistic term sets as well as the process of rating a service with respect to the these terms sets. One experiment with respect to this is to evaluate if, for a fixed set of services or SaaS offerings, multiple domain experts come up with term sets and

ratings that are roughly equivalent. 2) For various sets of services, the quality of the ranking could be evaluated by domain experts as well. It is to see if the result of the ranking meets the expectations from domain experts and whether there are any surprises that they would not have expected and, thus, giving them new insights. Both aspects require a significant effort as an empirical evaluation is required.

Acknowledgements

We would like to thank Dr. David Llewellyn-Jones for his support and advices. This research is supported by The Natural Science Foundation of Liaoning Province (2013020022) "Hybrid multi-criteria group decision making with various forms of information expression", and Liaoning Education Science "Twelfth Five Year Plan".

References

[1] T. Erl, Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall PTR, 2005.

[2] M. Gudgin, M. Hadley, N. Mendelsohn, J. Moreau, H. Nielsen, A. Karmarkar, Y. Lafon, Simple object access protocol (soap) 1.2, Tech. rep., World Wide Web Consortium (W3C) (2007).

[3] H. Ludwig, A. Keller, A. Dan, R. P. King, R. Franck, Web service level agreement (wsla) language specification, version 1.0, Tech. rep., IBM Corporation (2003).

[4] A. Nadalin, M. Goodner, , M. Gudgin, A. Barbir, H. Granqvist, Ws-securitypolicy 1.2, `http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/ws-securitypolicy-1.2-spec-os.html`, [Online; accessed June-2015] (2007).

[5] S. Cantor, J. Kemp, R. Philpott, E. Maler, Xml signature syntax and processing (second edition), http://docs.oasis-open.org/security/saml/v2.0/saml-2.0-os.zip, [Online; accessed June-2015] (2008).

[6] D. Eastlake, J. Reagle, D. Solo, F. Hirsch, T. Roessler, Assertions and protocols for the oasis security assertion markup language (saml) 2.0, http://www.w3.org/TR/xmldsig-core/, [Online; accessed June-2015] (2008).

[7] D. Eastlake, J. Reagle, Xml encryption syntax and processing, http://www.w3.org/TR/xmlenc-core/, [Online; accessed June-2015] (2002).

[8] A. D. Brucker, F. Malmignati, M. Merabti, Q. Shi, B. Zhou, A framework for secure service composition, in: PASSAT, IEEE Computer Society, 2013, pp. 647–652. doi:10.1109/SocialCom.2013.97.

[9] L. Zadeh, A computational approach to fuzzy quantifiers in natural languages, Computers and Mathematics with Applications 9 (1) (1983) 149–184.

[10] F. Herrera, E. Herrera-Viedma, Linguistic decision analysis: steps for solving decision problems under linguistic information, Fuzzy Sets and Systems 115 (1) (2000) 67–82.

[11] W. Pedrycz, Why triangular membership functions?, Fuzzy Sets and Systems 64 (1) (1994) 21–30.

[12] Q. Zhang, Y. Sun, H. Yuan, Group decision evaluation on the quality of pure electronic journals, in: 2012 Internal Conference on Systems and Informatics (ICSAI 2012), 2012, pp. 810–814.

[13] Z. Xu, Uncertain linguistic aggregation operators based approach to multiple attribute group decision making under uncertain linguistic environment, Information Sciences 168 (1-4) (2004) 171–184.

[14] Z. Xu, Evaluation linguistic terms based approach to multiple attribute group decision making, System Engineering 20 (1) (2005) 84–88.

[15] F. Herrera, L. Martinez, P. Sanchez, Managing non-homogeneous information in group decision making, European Journal of Operational Research 166 (1) (2005) 115–132.

[16] T. L. Saaty, How to make a decision: The analytic hierarchy process, European Journal of Operational Research 48 (1) (1990) 9–26.

[17] Amazon, Aws security features, https://aws.amazon.com/security/aws-security-features/, [Online; accessed June-2015].

[18] Dropbox, Your stuff is safe with dropbox, https://www.dropbox.com/security/, [Online; accessed June-2015].

[19] Google, Google cloud platform security, https://cloud.google.com/security/, [Online; accessed June-2015].

[20] Microsoft, Microsoft azure trust center: security, http://azure.microsoft.com/en-gb/support/trust-center/security/, [Online; accessed June-2015].

[21] R. Henning, Security service level agreements: Quantifiable security for the enterprise?, in: Workshop on New Security Paradigm, 2009, pp. 54–60.

[22] M. Hale, R. Gamble, Secagreement: Advancing security risk calculations in cloud services, in: IEEE World Congress on Services, 2012, pp. 133–140.

[23] M. Hale, R. Gamble, Building a compliance vocabulary to embed security controls in cloud slas, in: IEEE Ninth World Congress on Services (SERVICES), 2013, pp. 118–125.

[24] E. Al-Masri, Q. Mahmoud, Qos-based discovery and ranking of web services, in: 16th International Conference on Computer Communications and Networks (ICCCN), 2007, pp. 529–534.

[25] H. Chan, T. Chieu, T. Kwok, Autonomic ranking and selection of web services by using single value decomposition technique, in: IEEE International Conference on Web Services (ICWS), 2008, pp. 661–666.

[26] D. Skoutas, D. Sacharidis, A. Simitsis, V. Kantere, T. Sellis, Top-k dominant web services under multi-criteria maching, in: 12th International Conference on Extending Database Technology: Advances in Database Technology (EDBT), 2009, pp. 898–909.

[27] G. D. Modica, G. Petralia, O. Tomarchio, A business ontology to enable semantic matchmaking in open cloud markets, in: 8th International Conference on Semantics, Knowledge and Grids (SKG), 2012, pp. 96–103.

[28] C. Irvine, T. Levin, Quality of security service, in: 2000 workshop on New security paradigms (NSPW), 2001, pp. 91–99.

[29] V. Casola, A. Fasolino, N. Mazzocca, P. Tramontana, An ahp-based framework for quality and security evaluation, in: 12th IEEE international conference on computational science and engineering (CSE), Vol. 3, 2009, pp. 405–411.

[30] C. Redl, I. Breskovic, I. Brandic, S. Dustdar, Automatic sla matching and provider selection in grid and cloud computing markets, in: 13th ACM/IEEE International Conference on Grid Computing (GRID), 2012, p. 8594.

[31] F. Chiclana, F. Herrera, E. Herrera-Viedma, Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations, Fuzzy Sets and Systems 97 (1) (1998) 33–48.

[32] J. Kacprzyk, Group decision making with a fuzzy linguistic majority, Fuzzy Sets and Systems 18 (2) (1986) 105–118.

[33] F. Chiclana, J. T. Garcia, M. del Moral, E. Herrera-Viedma, A statistical comparative study of different similarity measures of consensus in group decision making, Information Sciences 221 (2013) 110–123.

[34] Z. Xu, X. Cai, Group consensus algorithms based on preference relations, Information Sciences 181 (1) (2011) 150–162.