ServiceTrust: Supporting Reputation-Oriented Service Selection

Qiang He[1,2], Jun Yan[3], Hai Jin[1], and Yun Yang[2]

[1] School of Computer Science and Technology
Huazhong University of Science and Technology, Wuhan, 430074, China
hjin@hust.edu.cn
[2] Faculty of Information and Communication Technologies
Swinburne University of Technology, Melbourne, Australia 3122
qhe@ict.swin.edu.au, yyang@swin.edu.au
[3] School of Information Systems and Technology
University of Wollongong, Wollongong, Australia 2522
jyan@uow.edu.au

Abstract. Service transactions, although attached with service level agreements, may still fail due to various reasons, intentionally or accidentally, in the open and volatile service-oriented environment. In service selection, consumers often need to estimate the trustworthiness of the provider with limited prior experience and knowledge about them. Moreover, the service-oriented environment exposes consumers to unique threats including malicious reputation manipulation and quality-of-service (QoS) abuse. This paper presents ServiceTrust – a novel trust management approach to support reputation-oriented service selection by quantifying and comparing the trustworthiness of providers based on their historic performance over service transactions. ServiceTrust combines a consumer's and other consumers' personal trust to estimate the provider's trust value. Our experimental results demonstrate that ServiceTrust can significantly increase the success rate of service transactions and is effective in resisting malicious reputation manipulation and QoS abuse.

Keywords: Service-oriented computing, Web services, service selection, trust, service reputation.

1 Introduction

Service-oriented computing (SOC) has been attracting tremendous attention from both the academic and industrial communities. Using SOC, various services across a spectrum of domains can be provided to service consumers over the Internet. Service consumers can look for preferred and qualified services through service registries, invoke services in a loosely coupled manner, and receive desired outcomes from invoked services. Moreover, services from distributed locations can be composed to create new value-added composite services. In the service-oriented environment, services are essentially considered merchandises so that *service level agreements*

L. Baresi, C.-H. Chi, and J. Suzuki (Eds.): ICSOC-ServiceWave 2009, LNCS 5900, pp. 269–284, 2009.

(SLAs) can be established between service consumers and providers to specify mutually-agreed understandings and expectations of the *quality-of-service* (QoS) [10].

However, service providers would not always successfully enforce the SLAs due to various reasons. SLA violations occur from time to time, intentionally or accidentally. For example, malicious service providers may strategically fail service transactions despite of the penalty specified in the previously established SLA. Service providers' failures to enforce SLAs may result in unpredictable consequences and noncompensable loss which cannot be specified in SLAs beforehand. In service selection, the QoS can be negotiated over, but the success rate of the service transaction cannot be provided by the service providers. This problem is especially severe in the service composition scenarios where the composite services are composed of several component services. The failure of an individual component service in this scenario may result in exceptions in the composite service. When searching for service providers, service consumers usually prefer those who are most likely to successfully enforce the SLAs.

In addition, the open and volatile service-oriented environment exposes service consumers to various threats. A widely recognised one is that malicious service providers manipulate service consumers to report incorrect feedbacks in order to boost their reputations or to ruin their competitors' reputations [13]. Another major threat is QoS abuse, where service providers strategically alter their QoS offering behaviour and then provide fraudulent services in order to earn profits [24].

Due to the above issues, in service selection, solutions should be provided to help the service consumers estimate the trustworthiness of the service providers, as suggested but not specified by [1, 12, 28]. However, it is difficult for a service consumer to determine how much it can trust a service provider due to the lack of sufficient experience and knowledge about the service provider. A direct approach to address this issue is to use a reputation system which collects and processes feedback about service providers' past behaviour [11, 18, 20, 25]. To the best of our knowledge, no reputation system has been tailored for service selection in the service-oriented environment and the threats described earlier have not been properly addressed.

Furthermore, the service providers in the service-oriented environment usually have unique identifications in order to allow the service consumers to identify their services. In contrast, the peers in the P2P environment are usually anonymous. This feature makes it difficult to stimulate the peers to develop and maintain long-term reputations. Therefore, existing trust and reputation systems in the P2P environment, which usually put a lot of effort in maintaining peers' anonymity property, are somehow unsuitable to be directly applied in the service-oriented environment where long-term reputation is desirable.

This paper proposes ServiceTrust, a novel reputation-based trust approach which supports reputation-oriented service selection by estimating service consumers' trust over service providers based on their historic performance for SLA enforcement. ServiceTrust can improve the success rate of service transactions by helping service consumers identify trustworthy service providers in the open and volatile service-oriented environment. Through analysing service providers' long-term performance, ServiceTrust can effectively resist malicious reputation manipulation. In addition, ServiceTrust can effectively resist QoS abuse by calculating transactional trust in consideration of the QoS of the past successful service transactions that a service

provider has performed. ServiceTrust is independent of the underlying communication model so that it can be applied to different distributed computing architecture such as client-server and P2P.

The rest of the paper is organised as follows. Section 2 analyses the requirements of a reputation-oriented trust management approach for the service-oriented environment. Section 3 introduces the ServiceTrust mechanisms. After that, section 4 demonstrates the performance of the proposed ServiceTrust mechanisms with experimental evaluation. Section 5 introduces the major related work, and finally, section 6 summarises the key contribution of this paper and outlines the future work.

2 Requirements Analysis

To design a trust approach that supports reputation-oriented service selection, the following two threats that exist in the service-oriented environment must be addressed.

Malicious reputation manipulation. Malicious service providers may manipulate service consumers through techniques such as bribery to provide incorrect ratings in order to boost their reputations or to ruin their competitors' reputations. Malicious service providers can also inject incorrect ratings by faking service consumers.

QoS abuse. Malicious service consumers and providers may strategically alter their behaviour in QoS offering in order to obtain profits. For example, malicious service providers may use successful service transactions with small amounts to obtain service consumers' trust and then defraud the service consumers of their money with fraudulent service transactions with large amounts. Genuine service providers may also strategically alter their behaviour under certain circumstances, e.g. given an order of a service transaction with an unusually large amount; a genuine service provider might make the transition into being a malicious service provider and then provide a fraudulent service transaction.

To resist the threat of malicious reputation manipulation, service consumers' trust over service providers should be built on service providers' long-term reputations which are evaluated based on service providers' long-term performance. Long-term reputations can smooth out short-term fluctuations and highlight long-term trends of service providers' reputations. Another benefit of basing service consumers' trust on service providers' long-term reputation is that it encourages service providers' trustworthy and consistent behaviour at present.

To resist the threat of QoS abuse, when evaluating service consumers' trust for individual service transactions, namely *transactional trust*, the QoS of the past successful service transactions that the service providers have performed must be taken into account. By doing so, potential fraudulent service transactions can be identified and avoided.

3 ServiceTrust Mechanisms

The reasons why long-term reputation can help the service consumer with evaluating the trustworthiness of the service providers are twofold. First, service consumers can

obtain information to estimate service providers' abilities to successfully perform the forthcoming service transactions. Second, service providers' expectation of long-term reputations creates an incentive for their good performance at present. In this section, we will introduce a hierarchical trust structure which consists of *local transactional rating*, *local trust*, *global trust* and *transactional trust*, and the supporting mechanisms.

3.1 Generating Local Transactional Ratings

A local transactional rating describes a service consumer's experience of an individual service transaction with a service provider. Some early works [11, 25, 27], which use binary rating systems for calculating peers' reputations, prove that binary-value ratings work well for file-sharing systems, in which a file is either a complete or an incomplete version. An SLA in the service-oriented environment can be seen as an equivalent of a file in a file-sharing system because an SLA also only has two finalised status: fulfilled or unfulfilled, representing a successful service transaction or a failed one. Using binary values to rate service transactions is simple and does not require service consumers' physical participation. Another advantage of adopting binary-value ratings is that the ratings are explicit – a service transaction is either successful or unsuccessful in fulfilling the attached SLA. However, some recent works [23, 26] adopt numeric rating systems, in which the ratings are in a certain interval, e.g. [0, 1]. Compared to binary-value ratings, numeric-value ratings can model more accurately a service consumer's experience of a service transaction. But it requires service consumers' direct participation in the rating process which might become an obstacle to the extensive use of the application. Moreover, service consumers' lack of incentive and knowledge to report authentic and accurate ratings over service transactions may result in undesired, inaccurate or even incorrect ratings.

To give application developers flexible choices, ServiceTrust supports both binary-value and numeric ratings. For binary-value ratings, *0* represents a failed service transaction and *1* represents a successful one. The definition of service consumer *i*'s local transactional rating over the n^{th} service transaction with service provider *j*, denoted as $r_{i,j}^{(n)}$, is defined as follows:

$$r_{i,j}^{(n)} = \begin{cases} 0 & \textit{service transaction failed} \\ 1 & \textit{service transaction succeeded} \end{cases} \tag{1}$$

Service consumer can also rate service transactions using a value in the interval of [0, 1], with 0 and 1 representing complete dissatisfaction and complete satisfaction respectively. Considering that service consumers might lack the knowledge of QoS satisfaction, it is advisable for application developers to provide the service consumers with necessary assistance in the rating process.

3.2 Aggregating Local Transactional Ratings

To obtain a service consumer's local trust over a service provider, local transactional ratings generated from the service consumer's past service transactions with the service provider need to be aggregated. In the aggregation, we consider the temporal

dimension when evaluating the credibility of the local transactional ratings. It is not only their values that matter, but also at what time they are recorded – we assume that the local transactional ratings are recorded upon the completion of the service transactions. The credibility of a local transactional rating diminishes as time elapses. The ratings over a service consumer's recent service transactions with a service provider are more credible than the old ones. Also, when combining a service consumer's and other service consumers' personal local trust (as detailed in Section 3.3), the recent ratings provided by one service consumer are more credible than the old ones provided by another service consumer.

We use exponential moving average (EMA) scheme [3] to aggregate a service consumer's local transactional ratings over a service provider. Weights are computed to represent the credibility of the ratings according to how old the ratings are. The weight of each older rating decreases exponentially, giving more credibility to recent ratings whilst not entirely discarding older ratings. By doing so, short-term fluctuation of ratings can be smoothed out and long-term trend can be highlighted. Since the threshold between short-term and long-term is application specific, ServiceTrust uses parameter θ, as a time window, to specify valid ratings when aggregating the local transactional ratings. Ratings lying outside of θ are considered obsolete and thus discarded in the aggregation. θ can be set accordingly by the application developers to meet the requirements of applications.

The elapsed time since a service transaction has been performed is used to express how old the corresponding rating is. In order to compute the elapsed time of the ratings, ServiceTrust requires the rating time, i.e. the time when the transaction is rated, to be recorded along with the rating in the form of 2-tuple: $(r_{i,j}^{(n)}, t_{i,j}^{(n)})$.

The process of calculating service consumer i's local trust over service provider j by aggregating the series of local transactional ratings over the past service transactions between them, i.e. $[(r_{i,j}^{(1)}, t_{i,j}^{(1)}), (r_{i,j}^{(2)}, t_{i,j}^{(2)}), ..., (r_{i,j}^{(n)}, t_{i,j}^{(n)})]$, consists of the following five steps.

1. Compute the elapsed time, denoted as $et_{i,j}^{(n)}$, since each transaction was rated. The series of local transactional ratings becomes: $[(r_{i,j}^{(1)}, et_{i,j}^{(1)}), (r_{i,j}^{(2)}, et_{i,j}^{(2)}), ..., (r_{i,j}^{(n)}, et_{i,j}^{(n)})]$;
2. Determine the value of the time window, θ;
3. Divide the time frame confined by θ into s time slots;
4. Compute the arithmetic average value of the local transactional ratings in each time slot, denoted as $ar_{i,j}^{(1)}, ar_{i,j}^{(2)}, ..., ar_{i,j}^{(s)}$;
5. Aggregate $ar_{i,j}^{(1)}, ar_{i,j}^{(2)}, ..., ar_{i,j}^{(s)}$ to obtain service consumer i's local aggregated rating over service provider j, denoted as $R_{i,j}$, using exponential averaging scheme as follows:

$$RR_{i,j} = \sum_{k=1}^{s} \alpha(1-\alpha)^k ar_{i,j}^{(k)} \tag{2}$$

where $0 < \alpha < 1$ controls how fast the credibility of the ratings decreases over time.

Besides θ, two other parameters, s and α, are manoeuvrable. They can be set by application developers to control the weight decrease in order to meet application

specific requirements. The bigger s and α are, the faster the weight decreases, meaning the faster the old ratings in θ become incredible.

3.3 Combining Personal Trust

The local trust introduced in Section 3.2 reflects a service consumer's personal opinion of a service provider. To comprehensively evaluate a service consumer's global trust over a service provider, the service consumer's local trust should be combined with other service consumer's local trust. By doing so, the service consumer can obtain a global and comprehensive view of the service provider. A simple approach to the combination is to simply average all the local trust. An advanced approach is to compute a weighted average of all the local trust, where the weights represent the credibility of the local trust.

The credibility of a service consumer's local trust over a service provider depends not only on how old the local transactional ratings are (see Section 3.1), but also on how long the service consumer has had interactions with the service provider. Experience with the service provider in the longer-term gives the service consumer more information and knowledge about the service provider, thus enabling the service consumer to predict the service provider's ability and behaviour better [6, 21]. It also provides a firmer basis for calculating the credibility of the service consumer's local trust over the service provider. Therefore, when incorporating other service consumers' local trust into evaluating a service consumer's global trust over a service provider, we consider the relationship duration between the service consumers and the service provider, measured by the number of past service transactions between them. The longer relationship duration a service consumer has with the service provider, the more credible its local trust over the service provider is.

We adopt Rayleigh cumulative distribution functions [19] to calculate the weights according to the number of a service consumer's past service transactions with the service provider. The credibility of service consumer i's local trust over service provider j, denoted as $\beta_{i,j}$, is calculated as follows:

$$\beta_{i,j} = 1 - exp(\frac{-x^2}{2\sigma^2}) \quad (\sigma > 0) \tag{3}$$

where σ is a parameter that inversely controls how fast $\beta_{i,j}$ increases as the number of interactions, denoted as x, increases. σ can be set by the application developers, from 0 to theoretically ∞, to capture the characteristics of different application scenarios.

Compared to other service consumers' local trust, a service consumer can choose to trust its own local trust more or less when evaluating its global trust over the service provider. To reflect this nature, the weight assigned to the service consumer's own local trust over the service provider, denoted as $\beta'_{i,j}$, is computed as follows:

$$\beta'_{i,j} = 1 - exp(\frac{-x^2}{2(\sigma + \varepsilon)^2}) \quad (\sigma + \varepsilon) > 0 \tag{4}$$

where x is the number of service transactions that service consumer i has had with service provider j and ε specifies *how much more* (using a negative number) or *how*

much less (using a positive number) the service consumer trusts its own local trust over service provider *j* than other service consumers'.

Then service consumer *i*'s global trust over service provider *j,* denoted as $\tilde{R}_{i,j}$, can be calculated as follows:

$$\tilde{R}_{i,j} = \beta'_{i,j} \cdot R'_{i,j} + \sum_{k} \beta_{k,j} \cdot R_{k,j} \tag{5}$$

where $R'_{i,j}$ is service consumer *i*'s own local trust over service provider *j* and $R_{k,j}$ is the k^{th} other service consumer's local trust over service provider *j*.

3.4 Evaluating Transactional Trust

The scheme presented in this section can be applied to prevent various types of QoS abuse, e.g. execution time, availability and throughput, etc. Since transaction amount is usually one of a service consumer's most important concerns about the service in the service-oriented environment, we present the solution to transaction amount abuse for demonstration.

To prevent service consumers from transaction amount abuse, we incorporate the transaction amount into estimating service consumers' transactional trust for individual service transactions. We define transactional trust as the probability at which a service consumer believes the service provider will perform an individual service transaction and deliver expected outcomes specified in the attached SLA.

Transaction amount abuse usually consists of two steps. First, the malicious service provider fulfils service transactions with relatively small amounts in order to obtain a service consumer's trust. Second, the malicious service provider entices the service consumer to give it an order for a service transaction with a large amount, and then defrauds the customer with fraudulent service transactions or inferior goods afterwards. Under other circumstances, a fraudulent service transaction might also be performed, e.g. a genuine service provider may make the transition into being malicious when it gets an order for a service transaction with an unusually large amount which reaches or crosses its threshold for being genuine.

We address this issue by evaluating the transactional trust in consideration of the similarity between the quote on the forthcoming service transaction and the average transaction amount of the successful service transactions the service provider has performed. The base for this approach is the spirit of situational trust [15]: experience from situations of a similar nature will give a means of determining risk accurately. When evaluating the transactional trust, we consider two factors:

1. The average amount of successful service transactions that the service provider has performed. In general, the larger the quote on a service transaction is than the average amount of its past successful service transactions, the more likely that the service provider will provide a fraudulent service transaction.
2. The extent of amounts of successful service transactions that the service provider has performed. If a service provider has a large extent of amounts of successful service transactions, the chance that it will provide a fraudulent service transaction is slim.

Combining the considerations on the above two factors, we evaluate service consumer *i*'s transactional trust for a forthcoming service transaction provided by service provider *j*, denoted as $\overline{R}_{i,j}$, using formula (6).

$$\overline{R}_{i,j} = \gamma \cdot \tilde{R}_{i,j} \tag{6}$$

$$\gamma = (\frac{1}{\Delta^2})^k \tag{7}$$

$$\Delta = \frac{q_{new}}{a_j^{ave}} \cdot \frac{1}{cv_j} \tag{8}$$

$$cv_j = \frac{\sqrt{\sum_{m=1}^{M} (a_j^m - a_j^{ave})^2}}{a_j^{ave}} \tag{9}$$

where γ is the *transactional amount impact factor*, k is the parameter that controls how fast the transactional trust decreases as Δ increases, q_{new} is the quote on the forthcoming service transaction, a_j^{ave} is the average amount of the successful service transactions provider j has performed, a_j^m is the amount of the m^{th} successful service transaction provider j has performed, and cv_j is the coefficient of variation of $a_j^1, a_j^2, ..., a_j^m, ..., a_j^M$. Parameter k can be set by application developers according to the requirements of the applications. For example, in the scenario where the fluctuation of prices is relatively violent, such as the global crude oil market, a small k is advisable.

Usually the smaller the transaction amount is, the better it is for the service consumers. However in relation to some QoS such as availability and throughput, the higher the better it is for the service consumers. In those cases, formula (10) can be used to replace formula (8):

$$\Delta = \frac{a_j^{ave}}{q_{new}} \cdot \frac{1}{cv_j} \tag{10}$$

3.5 Initial Trust for New Services

In the discussion so far, we assume that a service provider provides one type of service. However, in the service-oriented environment, a service providers might be able to provide multiple types of services with respective service identifications. Accordingly, in ServiceTrust, a service consumer's trust over a service provider is service specific, and is estimated based on the service provider's historic performance over an individual type of services. It is possible that when a service provider starts offering a new service, there is no historic performance information about the new service for service consumers to refer to. In this case, a service consumer's trust for this new service cannot be evaluated as described above.

The development of a service consumer's initial trust for a new service usually goes through two stages: an exploratory stage and a commitment stage, which reflect

the general belief in the trust literature [2]. At the exploratory stage, the service provider's reputation will influence the service consumer's intention to trust the service provider. At the commitment stage, experience-based knowledge will readily replace the tentative trust built at the exploratory stage [16]. Another factor that influences a service provider's tentative trust over a service provider is its familiarity with the service provider [7, 14]. Familiarity is referred to as the understanding of the context which the service transaction is involved, and hence is considered the precondition for tentative trust [14].

From the perspectives of both reputation and context, we assume that a service provider with good reputation obtained from its existing services tends to provide the new service at a high success rate. This assumption is acceptable at least at the early stage of the new service's appearance because the service provider has to cater for the service consumers in order to quickly develop its reputation for the new service and to attract more potential service consumers [17]. Therefore, a service consumer's initial trust for a new service can be estimated through looking into the service provider's *global reputation* which is obtained by aggregating its reputations for its other services. And the estimation of a service consumer's initial trust for the new service is based on the service provider's global reputation. After interacting with the service provider, the service consumer can gradually incorporate its own experience and knowledge into developing its trust for the service following the procedure presented above (Sections 3.1-3.4). In ServiceTrust, service consumer *i*'s global trust over service provider *j*, denoted as $\hat{R}_{i,j}$, based on its trust for service provider *j*'s *N* individual existing services is calculated as:

$$\hat{R}_{i,j} = \frac{1}{N}\sum_{n=1}^{N} \tilde{R}_{i,j}^{(n)} \tag{11}$$

where $\tilde{R}_{i,j}^{(n)}$ is service consumer *i*'s trust for the n^{th} individual existing service provided by service provider *j*.

4 Experiments

In this section, we will assess the effectiveness of ServiceTrust as compared to a random service selection with no trust and reputation systems enabled. And then we will demonstrate our approach's resistibility against the threats of malicious reputation manipulation and QoS abuse. The issue of initial trust for new services is not directly related to either effectiveness on service selection or resistibility against threats and hence is not included in the experiments.

4.1 Experiments Configuration

Network model. We set up a service-oriented environment based on our previous work [8] in which peers look up each other in an efficient decentralised way**.** The simulation environment consists of 2000 service consumers and 200 service providers. Service consumers can request for services and service providers respond to these requests. Service consumers can access all the information about service providers' historic performance.

Node model. 20 types of services are provided by the 200 service providers, 10 for each. Each service provider has an inherent success rate randomly picked from a certain interval for its past and forthcoming service transactions. Different intervals for inherent success rates, including [0.9, 1], [0.8, 1] [0.7, 1], [0.6, 1], [0.5, 1] and [0.4, 1], are used to describe different volatile environments, [0.9, 1] being the best and [0.4, 1] being the worst. Throughout all experiments, service providers perform service transactions at their inherent success rates except under threat model #5. In the experiments with ServiceTrust enabled, genuine service consumers select the available service provider they have the highest trust over (global trust in experiments #1 to #5 and transactional trust in experiment #6), and rate service transactions honestly. Malicious service consumers select service providers and rate service transactions under corresponding threat models. The threat models are detailed in Table 1. In experiments where ServiceTrust is disabled, service consumers randomly select service providers.

Table 1. Threat models

Threat Models	Malicious Service Providers	Malicious Service Consumers	
		Service Selection	Rating
Threat Model #1	NA	randomly select service providers	rate 1 over all service transactions with malicious service providers
Threat Model #2	NA	select only malicious service providers	rate 1 over all service transactions with malicious service providers
Threat Model #3	NA	randomly select service providers	rate 0 over all service transactions with genuine service providers
Threat Model #4	NA	select only genuine service providers	rate over to all service transactions with genuine service providers
Threat Model #5	provide fraudulent services at the probability of $1 - \lambda$	NA	NA

ServiceTrust parameters. Table 2 summarises the parameters carefully chosen for the simulation in order to calculate service consumers' trust over service providers based on their historic performance in the long term.

Simulation execution. The simulation proceeds in simulation cycles. Each simulation cycle is subdivided into an evaluation cycle, a transaction cycle and a rating cycle. In an evaluation cycle, service consumers look up service providers and then evaluate

Table 2. ServiceTrust parameters used in simulation

α	θ	s	σ	ε	k
0.1	10 simulation cycles	10	15	-5	1/7

their global trust or transactional trust over the service providers. In a transaction cycle, each service consumer requests one service based on the results from trust evaluation in the evaluation cycle. Service providers correspond and complete service transactions. In each simulation cycle, each service provider can accommodate up to a maximum of 40 service consumers. If a service provider is fully loaded, the service consumer will turn to the service provider it has the next highest trust over. In a rating cycle, service consumers rate the service transactions honestly or under corresponding threat models. Binary rating values, described in Section 3.1, are used[1]. Upon the completion of each simulation cycle, statistics are collected at each service consumer. Each experiment is run 20 times and the results of all runs are averaged. We analyse the statistics to assess ServiceTrust by measuring the average success rates of overall service transactions.

4.2 Experimental Results

In experiment #1, we compare the average success rates of overall service transactions with ServiceTrust enabled against disabled in volatile environments without malicious service consumers and providers.

Figure 1 depicts results from experiment #1, showing that ServiceTrust can significantly increase the average success rates of overall service transactions in different volatile experiments. As the environment gets more volatile, the average success rate decreases drastically in the absence of ServiceTrust. However, with ServiceTrust enabled, even when different service providers' success rates vary in the large interval, i.e. [0.4, 1], the average success rate of overall service transactions still remains at 93%.

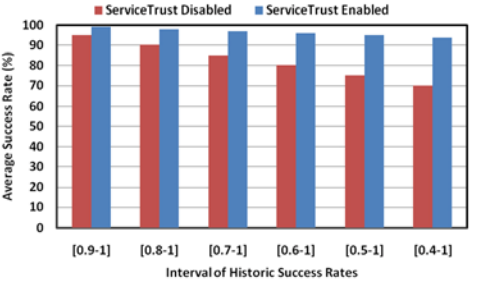

Fig. 1. Increase of average success rates of overall service transactions with ServiceTrust enabled

Then we conduct experiments #2-# 5 to evaluate ServiceTrust's resistibility against the threats of malicious reputation manipulation. Malicious reputation manipulation includes patterns described by four threat models: individual and collective malicious reputation boost, individual and collective malicious reputation ruin (threat models #1-#4). As shown in Figures 2-5, the experimental results demonstrate that

[1] We choose not to use numeric ratings to avoid unnecessary issue of modeling service consumer's satisfaction from QoS.

ServiceTrust can well protect the trust management from being undermined by these four threats in the long term.

Finally we test ServiceTrust's resistibility against the threats of QoS abuse (threat model #5) via experiment #6. This threat model describes the providers' strategic change of behaviour in QoS offering. We simulated the scenarios in which malicious service providers provide fraudulent services at the probability of $1-\gamma$. The results, as depicted in Figure 6, show that ServiceTrust can almost perfectly protect the consumers from being deceived by QoS abuse. In the most volatile environment with 70%

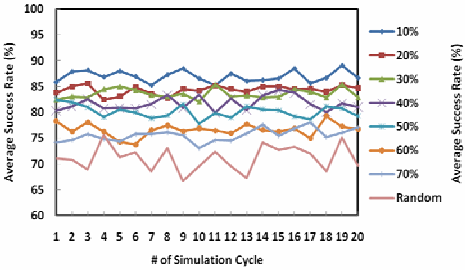

Fig. 2. Average success rates in different volatile environments under threat model #1

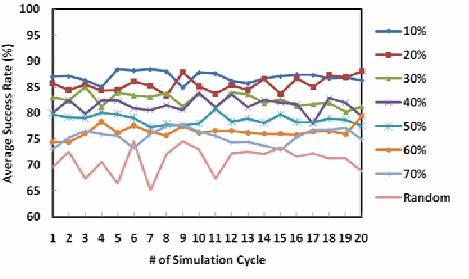

Fig. 3. Average success rates in different volatile environments under threat model #2

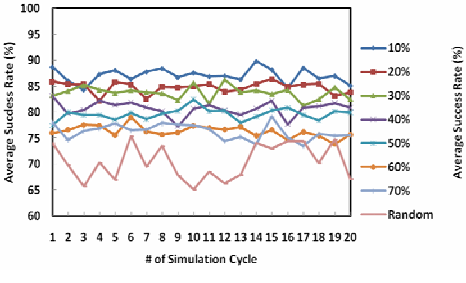

Fig. 4. Average success rates in different volatile environments under threat model #3

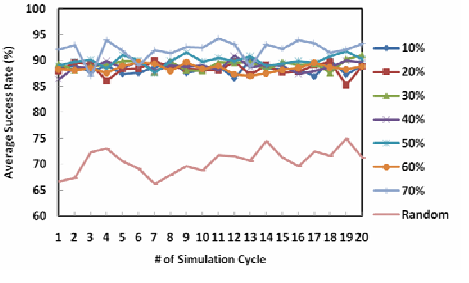

Fig. 5. Average success rates in different volatile environments under threat model #4

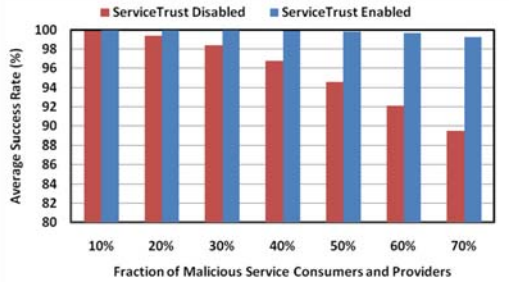

Fig. 6. Average success rate in different volatile environments under threat model #5

malicious service consumers and providers, the average success rate is still above 99%. The reason is that when the QoS is unusually better than the normal QoS that a service provider used to provide, the service consumer's transactional trust over that service provider drops immediately and drastically. The chance is very slim that a malicious service provider will be selected by a service consumer.

5 Related Work

Reputation-based trust research is being carried out in several distinct areas, most notably computer science and economics. An overview of many trust systems for online service provision can be found in [9]. And many key issues in reputation-based trust evaluation mechanisms in e-commerce environments are discussed in [22].

In the domain of distributed computing, several reputation systems have been proposed. Cornelli et al. [4] proposes P2PRep, a P2P protocol which complements Gnutella - an existing P2P file-sharing protocol. In P2PRep, peers can keep track of and share information about other peers' reputation. However, there are no formalised approaches to evaluate the reputation and credibility of the peers and no experimental evaluation is provided. Damiani et al. [5] enhance their previous work in [4] by introducing XRep, a distributed polling protocol that inquires the P2P network for peers' opinions (votes) on targeted resources. Votes are clustered based on IP address to prevent Sybil and collaboration attack. XRep focuses on supporting anonymous and secure services while preserving anonymity to a degree. Kamvar et al. [11] proposes EigenTrust, a distributed method for P2P file-sharing networks. Unique global trust values are computed and assigned to each peer in the network. EigenTrust requires pretrusted peers in the network to address the collusion problem. The limitation of their approach is that pretrusted peers may not always be available in all cases. Xiong et al. [25] proposes PeerTrust, a feedback based trust management system. PeerTrust incorporates three basic trust parameters (the feedback, the total number of transactions a peer performs and the credibility of the feedback sources) and two adaptive factors (transaction context factor and the community context factor) into computing the trustworthiness of peers. However, the solution adopted to measure feedback credibility, namely Trust-Value based credibility Measure (TVM), assumes that trustworthy nodes be more likely to be honest on the feedback they provide. This assumption is not generally true because peers may send incorrect feedbacks to ruin the reputations of its competitors. Srivatsa et al. [20] proposes TrustGuard, a safeguard framework in decentralised overlay networks, aiming at countering various vulnerabilities in reputation management. In TrustGuard, a peer rates credibility of feedback from other peers using a personalised similarity measure (PSM). Feedbacks that are similar to the peer's own are considered more credible. This method is limited in the cases where peers with long-term reputation are preferable and credible. For example, if a provider peer delivers a bad service transaction to a consumer peer by accident, malicious peers can flood bad feedbacks to rapidly ruin the consumer peer's trust over the provider peer.

Wang et al. [24] presents a model which incorporates transaction amount into trust evaluation. A simple method is proposed to measure the difference between old and

new transaction amounts. However, no amount-related malicious behaviour is modelled and no experimental results are presented to validate their approach.

Our work focuses on the crossroad of SOC, electronic ecommerce and distributed computing, and differs from the above works in a number of ways. First, ServiceTrust evaluates service consumers' trust over service providers based on their performance over past service transactions in the long term. Second, the temporal factor and relationship duration between a service consumer and a service provider are taken into account when evaluating the credibility of the service consumer's local trust of the service provider. Third, we address two unique and critical threats faced by reputation-based trust systems in the service-oriented environment, namely, malicious reputation manipulation and QoS abuse. Finally, we implement ServiceTrust and demonstrate the effectiveness of ServiceTrust on service selection, and the resistibility against the two threats.

6 Conclusions and Future Work

We have presented ServiceTrust – a novel trust management approach to support reputation-oriented service selection. The proposed approach aims at addressing unique threats in the service-oriented environment including malicious reputation manipulation and QoS abuse. In ServiceTrust, we evaluate a consumer's trust over a provider based on the provider's historic performance over service transactions in the long term. A consumer's local trust over a provider is combined with other consumers' to evaluate the consumer's global trust over the provider. The credibility of a consumer's local trust over a provider is calculated by considering the temporal factor and the relationship duration between the consumer and the provider. In order to resist QoS abuse, the comparison between the QoS of the forthcoming service transaction and the QoS of the successful service transactions that a provider has performed is taken into account when calculating a consumer's transactional trust. We have demonstrated experimental results which show that ServiceTrust can significantly improve the average success rate of service transactions by facilitating reputation-oriented service selection. In addition, experimental results show that ServiceTrust can well resist the following malicious threats: 1) individual malicious reputation boost; 2) collective malicious reputation boost; 3) individual malicious reputation ruin; 4) collective malicious reputation ruin; and 5) QoS abuse.

In the future, we will develop a complementary scheme to offer incentive to consumers to participate in ServiceTrust and provide correct ratings over service transactions. The resistibility against more threats will be further investigated in ServiceTrust.

Acknowledgement

This work is partly funded by the Australian Research Council Discovery Project Scheme under grant No.DP0663841, National Science Foundation of China under grant No.90412010 and ChinaGrid project from Ministry of Education of China. We are grateful for S. Hunter's help with conducting the experiments.

References

1. Ardagna, D., Pernici, B.: Adaptive Service Composition in Flexible Processes. IEEE Transactions on Software Engineering 33(6), 369–384 (2007)
2. Blau, P.: Exchange and Power in Social Life. John Wiley & Sons, New York (1964)
3. Chou, Y.-l.: Statistical Analysis: With Business and Economic Applications. Holt, Rinehart and Winston (1969)
4. Cornelli, F., Damiani, E., di Vimercati, S.D.C., Paraboschi, S., Samarati, P.: Choosing Reputable Servents in a P2P Network. In: Proceedings of 11th International Conference on World Wide Web, pp. 376–386. ACM Press, Honolulu (2002)
5. Damiani, E., Vimercati, D.C.D., Paraboschi, S., Samarati, P., Violante, F.: A Reputation-Based Approach for Choosing Reliable Resources in Peer-to-Peer Networks. In: Proceedings of 9th ACM Conference on Computer and Communications Security, pp. 207–216. ACM Press, Washington (2002)
6. Doyle, S.X., Roth, G.T.: Selling and Sales Management in Action: The Use of Insight Coaching to Improve Relationship Selling. Journal of Personal Selling & Sales Management 12(1), 59–64 (1992)
7. Gefen, D.: E-Commerce: the Role of Familiarity and Trust. Omega 28(6), 725–737 (2000)
8. He, Q., Yan, J., Yang, Y., Kowalczyk, R., Jin, H.: Chord4S: A P2P-based Decentralised Service Discovery Approach. In: Proceedings of IEEE International Conference on Services Computing, pp. 221–228. IEEE Computer Society, Honolulu (2008)
9. Jøsang, A., Ismail, R., Boyd, C.: A Survey of Trust and Reputation Systems for Online Service Provision. Decision Support Systems 43(2), 618–644 (2007)
10. Jin, L.-j., Machiraju, V., Sahai, A.: Analysis on Service Level Agreement of Web Services. Technical Report, HP Laboratories (2002), `http://www.hpl.hp.co.uk/techreports/2002/HPL-2002-180.pdf`
11. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The EigenTrust Algorithm for Reputation Management in P2P Networks. In: Proceedings of 12th International World Wide Web Conference, pp. 640–651. ACM Press, Budapest (2003)
12. Ko, J.M., Kim, C.O., Kwon, I.-H.: Quality-of-Service Oriented Web Service Composition Algorithm and Planning Architecture. Journal of Systems and Software 81(11), 2079–2090 (2008)
13. Lam, S.K., Riedl, J.: Shilling Recommender Systems for Fun and Profit. In: Proceedings of 13th International Conference on World Wide Web, pp. 393–402. ACM Press, New York (2004)
14. Luhmann, N.: Trust and Power. Wiley, Chichester (1979)
15. Marsh, S.P.: Formalising Trust as a Computational Concept, in Department of Mathematics and Computer Science Stirling, Scotland, UK, University of Stirling (1994)
16. McKnight, D.H., Choudhury, V., Kacmar, C.: Trust in E-Commerce Vendors: A Two-Stage Model. In: Proceedings of 21st International Conference on Information Systems, pp. 532–536. ACM Press, Brisbane (2000)
17. Mitchell, W.: Dual Clocks: Entry Order Influences on Incumbent and Newcomer Market Share and Survival When Specialized Assets Retain Their Value. Strategic Management Journal 12(2), 85–100 (1991)
18. Resnick, P., Kuwabara, K., Zeckhauser, R., Friedman, E.: Reputation Systems. Communications of the ACM 43(12), 45–48 (2000)
19. Ross, S.M.: Introduction to Probability and Statistics for Engineers and Scientists. Academic Press, Cleveland (2000)

20. Srivatsa, M., Xiong, L., Liu, L.: TrustGuard: Countering Vulnerabilities in Reputation Management for Decentralized Overlay Networks. In: Proceedings of 14th International Conference on World Wide Web, pp. 422–431. ACM Press, Chiba (2005)
21. Swan, J.E., Nolan, J.J.: Gaining Customer Trust: A Conceptual Guide for the Salesperson. Journal of Personal Selling & Sales Management 5(2), 39–48 (1985)
22. Wang, Y., Lin, K.-J.: Reputation-Oriented Trustworthy Computing in E-Commerce Environments. IEEE Internet Computing 12(4), 55–59 (2008)
23. Wang, Y., Varadharajan, V.: A Time-Based Peer Trust Evaluation in P2P E-commerce Environments. In: Zhou, X., Su, S., Papazoglou, M.P., Orlowska, M.E., Jeffery, K. (eds.) WISE 2004. LNCS, vol. 3306, pp. 730–735. Springer, Heidelberg (2004)
24. Wang, Y., Wong, D.S., Lin, K.-J., Varadharajan, V.: Evaluating Transaction Trust and Risk Levels in Peer-to-Peer E-Commerce Environments Information Systems and E-Business Management 6(1), 25–48 (2008)
25. Xiong, L., Liu, L.: PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities. IEEE Transactions on Knowledge and Data Engineering 16(7), 843–857 (2004)
26. Yu, B., Singh, M.P., Sycara, K.: Developing Trust in Large-Scale Peer-to-Peer Systems. In: Proceedings of 1st IEEE Symposium on Multi-Agent Security and Survivability, pp. 1–10. IEEE CS Press, Philadelphia (2004)
27. Yu, B., Singh, M.P., Sycara, K.: A Reputation-Based Approach for Choosing Reliable Resources in Peer to Peer Networks. In: Proceedings of 9th ACM Conference on Computer and Communications Security, pp. 207–216. ACM Press, Washington DC (2002)
28. Zeng, L., Benatallah, B., Dumas, M., Kalagnanam, J., Sheng, Q.Z.: Quality Driven Web Services Composition. In: Proceedings of 12th International Conference on World Wide Web, Budapest, Hungary, pp. 411–421 (2003)