

Limiting Sybil Attacks on Bayesian Trust Models in Open SOA Environments

Sebastian Ries, Erwin Aitenbichler

Telecooperation Group, Department of Computer Science, Technische Universität Darmstadt
Darmstadt, Germany

{ries,erwin}@tk.informatik.tu-darmstadt.de

Abstract

Next generation SOA systems promise to enable an “Internet of Services” - an open environment, in which every participant is not only free to offer and consume services, but also to provide central components of the ecosystem, such as marketplaces and community portals. In such an environment with no or only very little central control, malicious users may try to gain profits by providing low quality services. An approach to solve this issue is to introduce a trust or reputation system, enabling market participants to identify trustworthy parties. An open market does not have access restrictions to the market, hence such systems are likely to be exposed to Sybil attacks. In this paper, we provide a novel approach for aggregating recommendations in the face of Sybil attacks, which allow a single party to try to multiply the influence of its recommendation(s) by creating a high number of seemingly independent entities.

1. Introduction

Open Service Environments (OSEs) are large-scale distributed systems in which every participant is free to offer or consume software services, in absence of centralized control. As base technology for Future Business Value Networks (FBVNs), OSEs enable competitive and efficient service markets. OSEs are also highly relevant for mobile and Ubiquitous Computing, because such systems often only have intermittent network connectivity. Hence, they cannot rely on central components in the fixed Internet.

The absence of centralized control raises several issues regarding security and trust. In open environments without user verification mechanisms, where participating entities are usually represented by a pseudonym, a special threat comes from so-called Sybil attacks [12]. Even if pseudonyms cannot be forged, i.e., they uniquely refer to the user that created the pseudonym, pseudonyms are usually cheap. This allows single users or service providers to be represented by an arbitrary number of entities. In decentralized trust models that use recommendations to overcome the lack of direct evidence, Sybil attacks present a severe threat: Single parties can multiply their influence as recommenders by creating additional entities. In the following, we provide a novel approach to limit the influence of Sybil attacks by little

trusted recommenders in Bayesian trust models introducing a novel approach that considers trustworthiness and the rank of a recommender.

The remainder of the paper is structured as follows. Section 2 introduces our vision of open service environments. Section 3 then shows how users can be supported in general when selecting services. Section 4 presents our new approach for aggregating recommendations in face of Sybil attacks, which may be easily transferred to other Bayesian trust and reputation models. Finally, the paper is concluded in Section 5.

2. Open Service Environments

Service-oriented Architecture (SOA) is an architectural style that facilitates loose coupling of components, and consequently enables flexible selection and substitution of services. However, today’s SOA systems are rather closed. They are only used within the boundaries of an enterprise, or sometimes within conglomerates of enterprises with long-standing cooperations. To match the reality of Business Value Networks (BVNs), current systems must evolve towards open service environments. BVNs are defined as:

A BVN emerges from dynamic interactions of loosely-coupled organizations, which are legally distinct but economically interdependent, performing different value-creating roles (e.g., suppliers, distributors, service providers, infrastructure providers) that leverage their core competencies in order to flexibly craft optimum response to rapidly changing markets and customer demands. Value is created via dynamic exchanges of shared information and resources among these organizations engaged in complex and coevolving wherein dominant players can shape the network context [1].

The term ‘Future Business Value Networks’ inherits the BVN concept and stands for a conceptional framework which describes organization models with configurations of value adding collaborations within cooperative social networks among enterprises, (public) organizations, and individuals. A further characteristic is the aim to achieve a common set of goals enabled through the *Internet of Services* (or any other upcoming technology framework). FBVNs are motivated by the marching processes of outsourcing, tertiarisation, globalization, and technical innovation.

The basis for such an Internet of Services is currently developed in the large-scale Theseus Programme [1]. Building on the notion of a SOA, interacting software components can be loosely coupled and distributed over the Internet. However, *decentralized control* must be distinguished from *decentralized software components that are organized in a decentralized manner*. The Theseus/TEXO platform allows for a fully decentralized service provisioning, since service consumers and providers are communicating in a peer-to-peer manner. In addition, aspects like service hosting, development, monitoring, and load balancing do not require centralized control. However, the platform also has centralized-control entities, such as the service marketplace, authentication and authorization services, and the community portal.

This architecture solves several important problems related to security and trust. First, the marketplace provider verifies the identities and particulars of market participants. Consequently, participants cannot randomly create additional user accounts. Second, all market participants can consult the marketplace as a trusted third party for authorization and to verify user rights. Third, users can rate services they have used and thus provide recommendations for others. Recommendations are centrally stored on the community portal.

While this architecture still scales arbitrarily in a technical sense, it does not scale in an economical sense. A monopoly for a central service marketplace is very unlikely, since efficient markets demand competition and it could be considered as illegal. A second important application area is Ubiquitous Computing, because here applications are often faced with only intermittent connectivity to the fixed Internet. Consequently, such applications cannot always rely on centralized services. In the following, we assume a fully open service environment with the absence of any centralized control.

3. Service Selection

In an open service market, where anybody is allowed to offer services, it seems natural that there will be numerous offers for services providing the same functionality. Whenever a customer has the choice between multiple such services, the quality of a service becomes important. For example, a customer who looks for a service providing information about online shops offering goods for the cheapest price, may select an unreliable service provider, which, e.g., prefers certain online shops, or a service provider with a high response time. Even if the service providers offer information about the quality of their services in the service descriptions, the customer cannot rely on this information, as it is not an obligation [2].

3.1. General Approaches

According to [2] there are three ways to improve the customer's choice:

1) *Service Level Agreement (SLA)*: Customer and service provider may negotiate an agreement on the quality of the service (SLA). If the service provider does not comply to the agreed service level, there may be a penalty. A service level agreement comes with the cost of time and expense. Furthermore, it requires a common ontology for quality of service metrics.

2) *Monitoring the quality of services by a trusted third party*: This approach requires a trusted third party and monitors (or sensors) to measure the quality of the services, e.g. execution time, which might be costly and result in a big overhead in a dynamic environment with many leaving and upcoming services.

3) *Customer feedback*: Feedback from customers can be collected and published by a trusted third party. This approach already disburdens the trusted third party from monitoring the services itself. Furthermore, it brings the advantage of quality of service information that cannot be directly collected by monitors. Finally, the information can also be managed in a decentralized manner.

The advantages of the last approach may be the reason for the upcoming research to apply trust and reputation mechanisms to web services. This approach can be applied to business-to-business (B2B), business-to-consumer (B2C), and consumer-to-consumer (C2C) environments. According to [2] current research focuses on reputation and trust mechanisms relying on a trusted third party (central node) to collect and publish the customers' feedback information. Yet, decentralized systems, in which all entities store their feedback on their own and may provide this information to others on request, may be considered to be more scalable and more flexible, as they do not require a trusted third party. Additionally, they allow each party for fully controlling what happens with the created information as anyone can decide with whom the information is shared.

3.2. Evidence-based Trust Models

Customer feedback leads directly to evidence-based trust and reputation models, i.e., models that try to estimate the trustworthiness (or reputation) of a service provider based on the evidence that reflects the outcome of previous interactions of a customer with the service. One of the most prominent examples of such an approach is the feedback score that supports buyers when selecting sellers on eBay (www.eBay.com). Furthermore, there has been a number of proposals of trust and reputation models in research, e.g., centralized models [3]–[5], and distributed models [6]–[10].

Bayesian trust models, as presented in [4], [6]–[9], provide a sound and simple mechanism to update trust values

whenever new evidence is available, and in principle they also allow to include the subjective prior knowledge of a user. Especially, they allow to interpret trust as a subjective probability which fits the definition of trust provided in [11].

3.3. Recommendations: A Need and a Threat

In decentralized systems, direct evidence between the consumer and the available service providers may be rare. This will be especially true on open service platforms, where service providers and costumers can dynamically join and leave. In order to overcome this problem, decentralized trust models consider recommendations from third parties which may also be referred to as indirect evidence.

As recommendations influence the selection of an interaction partner, they provide a means for misuse. Recommenders might have a direct interest to improve the chances of a certain candidate to become selected, or to diminish the chances of others. Therefore, a main challenge is dealing with entities trying to provide misleading recommendations, either false accusations or false praise. A trust model needs appropriate mechanisms for filtering and weighting recommendations. For example, recommenders that have provided mostly misleading recommendations should have less influence (if any) than recommenders known for providing good recommendations.

In evidence-based trust models, the trust an entity has into another one, is usually linked to a pseudonym. In open environments without any access control this leads to the following situation: Even in the case that pseudonyms cannot be forged, i.e., they uniquely refer to the user that created the pseudonym, pseudonyms are usually cheap. As a consequence, single providers or customers can create several seemingly independent entities, which may be referred to as Sybil attack [12]. Thus, an attacker can try to multiply his influence as recommender by creating numerous seemingly independent entities, which collectively provide misleading recommendations. Because a typical assumption when aggregating recommendations is that they are based on independent observations, this attack is a severe threat to numerous trust models. In the following, a new approach to cope with Sybil attacks is presented. It should be easily transferable to other Bayesian trust models, e.g., [6], [7], [13].

4. Trust Model

This section starts with a very short introduction of a representation of trust which is based on the trust and reputation models presented in [4], [6]–[9]. Then, a basic computational model is introduced, and an extension of this basic mechanism which can be shown to be more robust to Sybil attacks.

4.1. Bayesian Representation

The main parameters used to derive the trustworthiness of an entity in the Bayesian representation are the numbers of positive r and negative s evidence that have been collected based on direct experience and recommendations. Furthermore, the parameters r_0 and s_0 reflect the prior knowledge. This is similar to [4], [6], [7], [9], [14]. The expectation value E is derived from the collected evidence and the prior knowledge using the parameters $\alpha = r + r_0$ and $\beta = s + s_0$ in the Beta distribution $Beta(\alpha, \beta)$.

$$E = \frac{r + r_0}{r + r_0 + s + s_0} \quad (1)$$

The corresponding Beta probability density function $f(p | \alpha, \beta)$ is defined as:

$$f(p | \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1} (1-p)^{\beta-1}, \quad (2)$$

where $0 \leq p \leq 1$, $\alpha > 0$, $\beta > 0$.

Given the parameters r_0 and s_0 , which may be chosen context-dependent as proposed in [9] or set to $r_0 = s_0 = 1$ [4], [6], [7]. The opinion about the trustworthiness of an entity is denoted as $o = (r, s)$. The expectation value of an opinion is referred to as $E(o)$ or $E((r, s))$.

4.2. Computational Model of Trust

In the following, we refer to the consumers and service providers as entities. The computational trust model provides means for combining the direct evidence of the initiator, i.e., the consumer that has to select a service, and recommendations by third parties. This can also be referred to as trust propagation. The basic ideas for trust propagation in the proposed approach are similar to the ones presented in [4], [14]. Therefore, the operators for the trust propagation are given the same names. The *consensus* operator combines several opinions to a single one, and the *discounting* operator allows to weight recommendations based on the opinion about the recommender.

For the explanation of the trust propagation a simple network is given as an example (Figure 1). Here, entity A is in the role of the initiator of an interaction, i.e., entity A has to select a service provider from a set of available service providers. As a basis for the selection, the initiator evaluates the trustworthiness of the candidates. In order to evaluate the trustworthiness of a candidate C , entity A uses its direct evidence and recommendations. In the example, entity A does not have any direct evidence, but it receives recommendations from the recommenders R_0, R_1, \dots, R_{100} .

From the example one can see that it is necessary to distinguish between the different contexts in which an entity gained trust. An entity may be trustworthy in the context

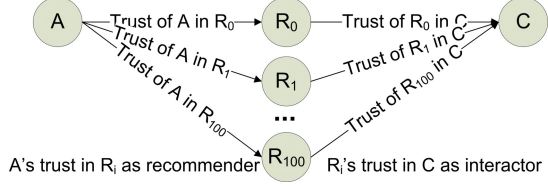


Figure 1. Trust network

of providing a service or in the context of providing recommendations about certain classes of service. The first is important for the selection of a candidate, the latter is important when deriving trust based on recommendations. An entity gains (dis-)trust as service provider, when it provides a satisfying quality of service. Yet, the behavior of an entity when providing a service does not necessarily convey information about its behavior as recommender – and vice versa – as both contexts refer to different capabilities of an entity. Therefore, trust is derived differently in both contexts. In the context of service provision, an entity gets trusted based on the quality of the service, and in the context of providing recommendations an entity gets trusted when providing accurate recommendations. In the following, we will assume that the trustworthiness of an entity in the context of providing recommendations is already given. For approaches to derive this information from the accuracy of previous recommendations see [7], [13].

Additionally, it is worth to mention that the trustworthiness in the context of providing recommendations as well as in the context of providing a service will also depend on the considered class of service, e.g., file sharing or weather forecast. Yet, in the following we assume there is only one class of service, in order to keep the notation simple. The opinion of an entity A about an entity B in the context of service provision is denoted as o_b^A (lowercase b), the opinion of an entity A about an entity B in the context of providing recommendations is denoted as o_B^A (uppercase B). Note, that the letter A will usually be used for the initiator of an interaction, the letters B and R for recommenders, and the letter C for the candidate. Furthermore, the expectation values $E(o_b^A)$ and $E(o_B^A)$ express the trustworthiness that entity A assigns to entity B in the context of interactions and recommendations, respectively.

4.3. Simple Trust Propagation

For trust propagation, we define two basic operators:

Definition 4.1 (Consensus): Let $o_c^A = (r_c^A, s_c^A)$ and $o_c^B = (r_c^B, s_c^B)$ be the opinions of A and B about the trustworthiness of entity c . The opinion $o_c^{A,B} = (r_c^{A,B}, s_c^{A,B})$ is modeled as the opinion of an imaginary entity which made the experiences of A and B , and is defined as:

$$o_c^{A,B} = o_c^A \oplus o_c^B = (r_c^A + r_c^B, s_c^A + s_c^B) \quad (3)$$

The ' \oplus ' symbol denotes the consensus operator. The operator can easily be extended for the consensus between multiple opinions (see Eq. 6).

Definition 4.2 (Discounting): Let $o_R^A = (r_R^A, s_R^A)$ and $o_c^R = (r_c^R, s_c^R)$. We denote the opinion of A about c based on the recommendation of R as o_c^{AR} and define it as:

$$o_c^{AR} = o_R^A \otimes o_c^R = (d(E(o_R^A)) \cdot r_c^R, d(E(o_R^A)) \cdot s_c^R) \quad (4)$$

The discounting factor $d(E(o_R^A))$ that defines the weight or the impact of the recommendation can be defined as:

$$d(E(o_R^A)) = \begin{cases} 0 & \text{if } E(o_R^A) \leq t_e \\ \frac{1}{1-t_e} \cdot (E(o_R^A) - t_e) & \text{else} \end{cases} \quad (5)$$

The ' \otimes ' symbol denotes the discounting operator.

Thus, the discounting factor increases with the trustworthiness of the recommender in the context of providing recommendations. Furthermore, the definition of the threshold t_e allows to exclude recommendations by recommenders with a trustworthiness below t_e by assigning a weight of 0. E.g., setting $t_e = 0.5$ allows to exclude recommendations from entities that provided mostly misleading recommendations, but still considers recommendations of entities that are unknown (as $E((0, 0)) = 0.5$ assuming $r_0 = s_0 = 1$).

Simple aggregation: In a simple case, the aggregation of recommendations is done using the operators defined above. Assume an entity A receives recommendations about a candidate C from a group of recommenders R_0, \dots, R_n . The trustworthiness that entity A assigns to R_0, \dots, R_n in the context of providing recommendations is given by $o_{R_0}^A, \dots, o_{R_n}^A$. The recommendations of the recommenders about the candidate are given as $o_c^{R_0}, \dots, o_c^{R_n}$. The aggregation of the opinions using the operators defined above is calculated as follows:

$$\begin{aligned} & (o_{R_0}^A \otimes o_c^{R_0}) \oplus \dots \oplus (o_{R_n}^A \otimes o_c^{R_n}) \\ &= \left(\sum_{i=0}^n d(E(o_{R_i}^A)) \cdot r_c^{R_i}, \sum_{i=0}^n d(E(o_{R_i}^A)) \cdot s_c^{R_i} \right) \end{aligned} \quad (6)$$

Whenever entity A has additional direct evidence, this evidence needs also be to considered, e.g., by adding this evidence after aggregating the recommendations or by considering oneself as recommender with trustworthiness 1.

4.4. Sybil-Attack-Resistant Trust Propagation

Although the approach introduced above considers the trustworthiness of a recommender, it does not consider whether the recommender is good or bad in relation to the rest of the available recommenders. In general, one could assume that the more highly trusted recommenders an entity has, the lower the influence of the less trusted entities

should be. This is reflected in the following approach that additionally considers the rank of a recommender. The most trustworthy recommender has a higher maximum influence than the second best recommender, and so on. This approach especially addresses two aspects:

1) It prevents that recommenders that are “lowly trusted”, but considered ($E(o_R^A) \geq t_e$), may provide an arbitrary high number of evidence in total.

2) It does not overly reduce the impact of recommendations by highly trusted recommenders.

The proposed solution is based on these ideas:

1) The maximum influence of recommendations increases with the trustworthiness of the recommender as above.

2) A new parameter *maximum number of recommendable evidence* N_R is introduced. N_R defines the maximum number of evidence that is considered per recommendation. Whenever a recommendation is based on a higher number of evidence, it will be normalized to a maximum of N_R units of evidence.

3) The maximum influence of a recommendation is limited based on the rank of its recommender. Additionally, the threshold t_s is introduced. The extended aggregation mechanism assures that all recommenders with a discounting factor below t_s are not able to provide in total more than N_R units of evidence to the aggregated opinion.

Whereas the first two aspects reduce the impact of a single recommender, the last one allows to take control on the aggregated impact of a group of recommenders.

The extended mechanism for aggregating recommendations is defined as follows: Assume (as above) an entity A receives recommendations about a candidate C from a group of recommenders R_0, \dots, R_n . The trustworthiness that entity A assigns to R_0, \dots, R_n in the context of providing recommendations is given by $o_{R_0}^A, \dots, o_{R_n}^A$. The recommendations about the candidate are given as $o_c^{R_0}, \dots, o_c^{R_n}$. In order to consider the rank of the recommenders, the recommenders need to be re-sorted, so that $E(o_{R_i}^A) \geq E(o_{R_{i+1}}^A)$ holds. After re-sorting the recommenders the extended aggregation mechanism is defined as:

$$\begin{aligned} & \left(\sum_{i=0}^n \min\{d(E(o_{R_i}^A)) \cdot r_c^{R_i}, \right. \\ & (1 - t_s) \cdot d(E(o_{R_i}^A))^i \cdot \frac{N_R}{r_c^{R_i} + s_c^{R_i}} \cdot r_c^{R_i}\}, \\ & \left. \min\{d(E(o_{R_i}^A)) \cdot s_c^{R_i}, \right. \\ & \left. (1 - t_s) \cdot d(E(o_{R_i}^A))^i \cdot \frac{N_R}{r_c^{R_i} + s_c^{R_i}} \cdot s_c^{R_i}\} \right) \end{aligned} \quad (7)$$

In general, this approach limits the maximum influence of a recommender R_i based on its rank i . Especially, it limits the total number of evidence $r_c^{R_i} + s_c^{R_i}$ that recommender R_i can provide at maximum to $(1 - t_s) \cdot d(E(o_{R_i}^A))^i \cdot N_R$. Thus, based on the properties of a geometric series the influence of an infinite number of recommenders with discounting factor

lower than t_s can be shown to be less than N_R (in case there are not any other recommenders available; otherwise, it is even lower). In contrast, the influence of recommenders with high trustworthiness (close to 1) still remains high, even if there are multiple such highly-trusted recommenders.

Example: Impact of the extended operator. The example is based on the following parameters: $N_R = 20$, $t_e = t_s = 0.5$, and $r_0 = s_0 = 1$. Here, entity A has to evaluate the trustworthiness of the service provider C . Entity A receives recommendations from R_0, \dots, R_{100} (see trust network in Fig. 1). Entity A 's trust in the recommenders in the context of providing recommendations $o_{R_i}^A$ and the recommendations $o_c^{R_i}$ by each recommender R_i are given in Table 1. In the table, the recommenders are already sorted according to their trustworthiness. The table shows three recommenders (R_0, R_1 , and R_2) that provided a higher number (13) of mostly accurate recommendations to entity A . Furthermore, it shows an attack on the trustworthiness of candidate C . The attacker created 97 Sybils R_3, \dots, R_{100} . In the past, each Sybil provided a single accurate recommendation in order to get considered in the evaluation of the trustworthiness of entity C . In the attack, each Sybil tries to provide a bad recommendation about entity C in order to reduce the calculated trust value.

In Table 1, one can see that the discounting factor $d(E(o_{R_i}^A))$ depends on the trustworthiness of the recommender. The value of X , which can be interpreted as the value of a discounted recommendation in the simple aggregation mechanism, does not consider the rank of the recommender, only its trustworthiness. Therefore, the recommendations by the attackers R_3, \dots, R_{100} have identical influence on the aggregated opinion. In contrast, the value Y additionally considers the rank of the recommender. Thus, the impact of recommender R_{100} on the aggregated opinion is significantly lower than the influence of R_3 .

Using the simple aggregation mechanism, the aggregated opinion would be (12.93, 654, 7). Here, the attacker outweighs the impact of the other recommenders. Using the improved recommendations mechanism leads to an aggregated opinion (11.38, 1.89). Here, the attack is not successful, as the aggregated opinion is clearly dominated by the recommendations of the more trusted recommenders.

When the attacking recommenders would have provided a higher number of negative evidence, their influence in the simple aggregation mechanism would even grow, whereas this would have (almost) no influence in the extended approach.

5. Conclusion

We provided a new computational model of trust. The main features of this model are:

i	0	1	2	3	4	5	...	100
$o_{R_i}^A$	(12, 1)	(11, 2)	(10, 3)	(1, 0)	(1, 0)	(1, 0)	...	(1, 0)
$o_c^{R_i}$	(6, 1)	(8, 1)	(8, 0)	(0, 20)	(0, 20)	(0, 20)	...	(0, 20)
$E(o_{R_i}^A)$	0.87	0.80	0.73	0.67	0.67	0.67	...	0.67
$d(E(o_{R_i}^A))$	0.73	0.60	0.47	0.33	0.33	0.33	...	0.33
X	(4.40, 0.73)	(4.80, 0.60)	(3.73, 0.00)	(0.00, 6.67)	(0.00, 6.67)	(0.00, 6.67)	...	(0.00, 6.67)
Y	(8.57, 1.43)	(5.33, 0.67)	(2.18, 0.00)	(0.00, 0.37)	(0.00, 0.12)	(0.00, 0.04)	...	(0.00, 1.9E - 47)
$\min(X, Y)$	(4.40, 0.73)	(4.80, 0.60)	(2.18, 0.00)	(0.00, 0.37)	(0.00, 0.12)	(0.00, 0.04)	...	(0.00, 1.9E - 47)

Table 1. Example: Sybil attack - it holds $X := (d(E(o_{R_i}^A)) \cdot r_c^{R_i}, d(E(o_{R_i}^A)) \cdot s_c^{R_i})$ and $Y := ((1 - t_s) \cdot d(E(o_{R_i}^A)))^i \cdot \frac{N_R}{r_c^{R_i} + s_c^{R_i}} \cdot r_c^{R_i}, (1 - t_s) \cdot d(E(o_{R_i}^A)))^i \cdot \frac{N_R}{r_c^{R_i} + s_c^{R_i}} \cdot s_c^{R_i}$

1) The discounting (weighting) of recommendations considers the trustworthiness of recommenders in the context of providing recommendations. Thus, the discounting is based on the right type of trust.

2) The influence of bad recommenders is reduced as recommendations by recommenders with an expectation value for providing accurate recommendations lower than t_e are excluded.

3) Recommendations by unknown recommenders can be considered in the case that the trust value of those recommenders is above t_e . This is important as in contexts in which one expects the recommendations by unknown entities to be accurate, those recommendations will be included in the aggregated opinion. Then, in absence of recommendations provided by trusted entities, recommendations by unknown entities can be a valuable contribution.

4) The aggregated opinion, which is derived from collected recommendations (and additionally direct evidence), favors the recommendations by the best recommenders. Thus, in presence of recommendations by highly trusted recommenders, the influence of recommendations by lower trust recommenders (potential attackers) is strongly reduced. Especially, the aggregation mechanism is robust to Sybil attacks in the sense that an attacker cannot contribute an arbitrary high amount of evidence to the aggregated opinion by simply increasing the number of recommenders. The novelty in this aggregation mechanism is that the trustworthiness and the rank of a recommender are considered in order to limit its maximal influence.

Finally, as the trust value can be interpreted as a probability, it can easily be integrated in decision making. Beyond simply choosing the best candidate available, the integration in utility-based decision making is possible.

Acknowledgments. This work was supported by the Center for Advanced Security Research Darmstadt (CASED) and by the Theseus Programme, funded by the German Federal Ministry of Economy and Technology under the promotional reference 01MQ07012.

References

[1] BMWi. (2009) TEXO - Business Webs in the

Internet of Services. [Online]. Available: <http://theseus-programm.de/scenarios/en/texo.html>

- [2] Y. Wang and J. Vassileva, "Toward trust and reputation based web service selection: A survey," *International Transactions on Systems Science and Applications (ITSSA)*, Vol 3, no. No. 2, 2007.
- [3] G. Zacharia, A. Moukas, and P. Maes, "Collaborative reputation mechanisms in electronic marketplaces," in *HICSS '99*, 1999.
- [4] A. Jøsang and R. Ismail, "The beta reputation system," in *Proceedings of the 15th Bled Conference on Electronic Commerce*, 2002.
- [5] A. Jøsang, X. Luo, and X. Chen, "Continuous ratings in discrete bayesian reputation systems," in *2nd Joint iTrust and PST Conference on Privacy, Trust Management and Security*, 2008, pp. 151–166.
- [6] S. Buchegger and J.-Y. Le Boudec, "A Robust Reputation System for Peer-to-Peer and Mobile Ad-hoc Networks," in *P2PEcon 2004*, 2004.
- [7] W. T. Teacy, J. Patel, N. R. Jennings, and M. Luck, "TRAVOS: Trust and reputation in the context of inaccurate information sources," *Autonomous Agents and Multi-Agent Systems*, vol. 12, no. 2, pp. 183–198, 2006.
- [8] S. Ries, "CertainTrust: A trust model for users and agents," in *ACM SAC*, 2007, pp. 1599 – 1604.
- [9] S. Ries, "Extending bayesian trust models regarding context-dependence and user friendly representation," in *ACM SAC*, 2009.
- [10] Z. Despotovic and K. Aberer, "Probabilistic Prediction of Peers' Performances in P2P Networks," *Engineering Applications of Artificial Intelligence*, vol. 18, no. 7, pp. 771–780, 2005.
- [11] D. Gambetta, "Can we trust trust?" in *Trust: Making and Breaking Cooperative Relations*, D. Gambetta, Ed. New York: Basil Blackwell, 1990, pp. 213–237.
- [12] J. R. Douceur, "The sybil attack," in *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*. 2002, pp. 251–260.
- [13] S. Ries and A. Heinemann, "Analyzing the robustness of CertainTrust," in *2nd Joint iTrust and PST Conference on Privacy, Trust Management and Security*, 2008, pp. 51 – 67.
- [14] A. Jøsang, "A logic for uncertain probabilities." *Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 9, no. 3, pp. 279–212, 2001.