Journal of Theoretical and Applied Electronic Commerce Research
ISSN 0718–1876 Electronic Version
VOL 5 / ISSUE 1 / APRIL 2010 / 1-15

This paper is available online at
www.jtaer.com

Addressing Common Vulnerabilities of Reputation Systems for Electronic Commerce

Yuan Yao[1,2], Sini Ruohomaa[3], and Feng Xu[1,2]

[1] State Key Laboratory for Novel Software Technology, Nanjing University, China.
[2] Department of Computer Science and Technology, Nanjing University, China.
YuanY@ics.nju.edu.cn, xf@nju.edu.cn
[3] Department of Computer Science, University of Helsinki, Finland.
sini.ruohomaa@cs.helsinki.fi

Abstract

Reputation systems provide a form of social control and reveal behaviour patterns in the uncertain and risk-laden environment of the open Internet. However, proposed reputation systems typically focus on the effectiveness and accuracy of reputation management, and suffer from a number of common vulnerabilities. As a result, introducing reputation management into the business environment may only replace the problems it hopes to solve with new issues. This paper aims to improve the security and robustness of reputation systems through 1) identifying the basic requirements in that area, 2) analyzing existing reputation systems for e-Commerce and a handful of other environments to compare their design choices and solutions provided, and 3) compiling a number of topical practices into guidelines for future research and development of reputation systems.

Key words: Reputation Systems, E-Commerce, Robustness and Security, Vulnerabilities, Attacks and Defenses

1 Introduction

Creating sustainable social environments on the Internet, for electronic commerce and other forms of collaboration, has changed the role of software in interactions. Securing systems against the uncertainty and risk in these open collaboration environments requires software-based support for social control, and reputation systems have been proposed to cater for this need [33, 57]. Reputation systems present expectations of a participant's future actions based on its past behaviour. These expectations can be used to support or to automate decisions on e.g. whether to collaborate with the participant, to engage in a commercial transaction with it, or to rely on information provided by it. Reputation-based trust systems utilize reputation information, but can include additional factors, such as business incentives, in the decisions to reflect a more general willingness to collaborate. In addition to helping people to decide whom to trust, reputation systems encourage trustworthy behaviour and deter dishonest participation [57].

While electronic marketplaces and e-Commerce form a major application area for reputation systems, notable reputation research has also been made in the context of collaboration in peer-to-peer networks and a number of other areas, such as routing in mobile ad hoc networks, or ensuring data accuracy, relevance and quality in different application environments [1, 4, 23, 53].

Security and robustness have been identified as pivotal challenges in the design and development of reputation systems [30, 37, 46, 47]. As the reputation system mediates trust between its users, it must itself be trusted to support good decisions through accurate and relevant information. What many proposed reputation systems have in common, however, is a number of basic vulnerabilities that appear to primarily result from lack of attention to a number of key requirements. While studies have been made on specific attacks against reputation systems already (e.g. [19, 27]), it is time to focus on addressing the larger-scale *vulnerabilities* that enable the different kinds of attacks. For that purpose, we define what the security and robustness requirements for reputation systems are, and identify some topical practices in implementing them from the literature. These solutions must in turn be set both in the context of the major design choices and goals of the systems applying them, and the context of the application areas, be it e-Commerce, peer-to-peer collaboration or something else. In this way, we can analyze the applicability of the presented solutions.

The rest of this paper is organized as follows: Section 2 presents related survey work. Section 3 defines six basic security and robustness requirements for reputation systems. Section 4 focuses on the design choice context of different solutions, presenting a decomposition of reputation systems into three dimensions: dissemination, calculation and evolution. Section 5 presents relevant attacks, as well as some proposed defense mechanisms for each of the six different vulnerabilities that follow from neglecting one or more of the requirements from Section 3. Section 6 discusses solutions for the requirements in the context of their application areas. Section 7 compiles a list of topical practices we have identified for addressing common vulnerabilities in reputation systems. Section 8 concludes the paper.

2 Related Work

In recent years, a number of surveys have been published on trust and reputation systems. Partly due to the cross-disciplinary nature and relative immaturity of the field, we have yet to see systematic literature reviews as defined by Kitchenham et al. [39]: the terminology is still quite mixed [62], making system categorization and comparison difficult, and publication forums are still evolving, meaning that publications representing considerable advances in the field are scattered across a number of conferences, workshops, journals and theses more often chosen for representing a specific application area than reputation systems. Our own literature review has been similarly ad hoc out of necessity, selecting a set of 20 reasonably well-known and widely cited systems from the years 2002-2010 for closer analysis. We extend the studied literature with proposals from a number of other studies that e.g. were not directly comparable to the example set due to solving different partial problems. We return to the selected systems in the next section.

Artz and Gil provide a survey on trust research from a broad perspective [6]. They categorize trust research into four classes, namely policy-based trust, reputation-based trust, general models of trust, and trust in information resources. Within this classification, our work focuses on reputation-based trust systems. We primarily distinguish between reputation systems and trust systems in that the former draw a decision primarily from reputation data, while a trust system is more general and may model various inputs, such as the business importance of the decision, or the requester's certificate-proven membership in a trusted group (e.g. [7]). In other words, trust systems measure the decision-maker's *willingness* to depend, while reputation systems measure an *expectation* of good outcomes based on information on the target actor's past behaviour. The former can influence the latter, which is why a number of reputation-based trust systems can be argued to belong to either category. Systems that only present the reputation information as a decision aid to a user, or as input to external decision systems, are most clearly distinguishable as pure reputation systems.

Marti and Garcia-Molina decompose reputation systems for peer-to-peer environments into three components: information gathering, reputation scoring and ranking, and action taking [48]. While they focus on desired properties of the three components separately, the paper also lists several attacks against reputation systems. In contrast, we focus on the e-Commerce environment, while including systems from other fields to broaden our analysis of different solutions. In relation to the decomposition, we find that reputation evolution is an important dimension that needs specific attention in our vulnerability analysis; our calculation dimension covers scoring and ranking as well as decision-making.

Sabater and Sierra focus on computational trust and reputation models and decompose them into seven dimensions of conceptual model, information sources, visibility types, models granularity, agent behavior assumptions, type of exchanged information, and trust/reputation reliability measure [67]. We cover these seven dimensions as aspects in our decomposition analysis. While Sabater and Sierra aim to provide an overview of current models and summarize them in detail, our focus is on proposing improvements to the security and robustness of reputation systems.

Jøsang et al. study the state of the art in trust and reputation systems [33]. Their work concentrates on live, online reputation systems in various application areas, and they also argue that current reputation systems are far from robust. Our work mainly considers reputation systems proposed in the literature, and presents guidelines for building more secure and robust reputation systems.

Ruohomaa et al. analyze the maturity of reputation systems from the point of view of how they support credibility analysis of reputation information [65]. The decomposition consists of recommendation creation and content, selection and use of recommenders, and reasoning and interpretation of the result. Credibility analysis is encompassed within one of the six security and robustness requirements we have identified, so the survey provides a more specialized study into one of our subtopics.

Hoffman et al. survey reputation systems from three points of view: through a decomposition framework dividing the systems to three dimensions of formulation, calculation and dissemination, through weakness against reputation manipulation and denial of service attacks, and through defense strategies against a few attack types [27]. The dimensions, weaknesses and defense strategies form three separate tracks in the paper. In contrast, our work binds common vulnerabilities to a set of security and robustness requirements, presenting attacks and proposed defenses in this context as well as the context of the application areas. In addition, we focus on more general guidelines than specific defenses against attacks: while it is valuable to know different behaviour patterns that can constitute a reputation attack, it is more fruitful to primarily address the underlying issues rather than concentrate effort to specialized defenses separately.

3 Robustness and Security Requirements

In this section, we define six security and robustness requirements for reputation systems. Failure to fulfil one or more of these basic requirements gives rise to the common vulnerabilities we present in more detail in Section 5. The severity of a vulnerability depends partially on the target application environment, which we will return to in Section 6. However, for any serious e-Commerce target application, all requirements must be met. The following six requirements can be divided into two categories: the first three tie into the reputation system on a lower level, while the last three tie into the computation of reputation scores and decision-making, i.e. the reputation metric.

Req. 1 - Message authenticity, integrity and confidentiality: As reputation systems are based on reputation information sharing, message security is integral to the functioning of the system. In the e-Commerce setting, this typically translates to message authenticity, integrity and confidentiality: if the credibility of the reputation information source is to be considered, it must be reliably known; the information must not be tampered with on the way; and, for limiting retaliation and reciprocity, the information must not be leaked to other actors than the ones specified in the system.

Req. 2 - Tamper-proofing and availability of stored reputation information: Reputation information can be stored either locally by the node making a decision, or externally, on a central server or distributed among a network of peers. In the case of external storage, it is essential to ensure that the information cannot be tampered with by third parties, and that it is available at decision time.

Req. 3 - Leverage-balanced identity management: Identity management must be set up in a way that prevents a specific actor from boosting their leverage in the reputation system without limit by creating new identities. There are two ways to approach this requirement: either by ensuring that multiple identities do not directly translate to increased leverage, or by balancing the cost of multiple identities with the leverage gained.

Req. 4 - Context-aware decision making: The reputation system should support making different decisions in different contexts, and be able to differentiate reputation information by the transactions it represents. While a reputation system can be implemented for an environment where all decisions and outcomes are genuinely equal in importance, in the e-Commerce setting this is seldom the case: there are transactions that have considerably higher value than others, or the suitability of a transaction partner for a specific task is not a given.

As the value-based attack described in Section 5.2.1 demonstrates, context-awareness is no longer simply a feature among others, but a matter of robustness.

Req. 5 - Recommendation credibility evaluation: In any distributed system, there are few elements that can be completely trusted. Human actors can lie, and automated monitors can produce incorrect data. Unintentional errors, such as misinterpretations and misunderstandings are possible as well. Shared reputation information can be incorrect for many reasons, and because of this, its credibility should always be evaluated in order to determine how much influence it should have on a given decision.

Req. 6 - Incongruity management in reputation evolution: Reputation information concerning past behaviour is used to try to predict future behaviour. However, the assumption on how these two are connected is sometimes violated, and the reputation system should be prepared to react to this kind of incongruity. As a simple example, the reputation system should be quick to drop the reputation of a previously well-behaved actor, if it begins to defect in its transactions. In the other extreme, an honest user's good reputation should be protected from misinformation, and one-time accidents should be treated with more forgiveness than consistent bad behaviour, unless they both actually indicate that the actor is no longer worth collaborating with. Naturally, in order to meet this requirement, a reputation system must somehow address reputation evolution, rather than considering it static; Section 4.3 discusses the topic further.

In the following section, we provide a context for these requirements by comparing the design choices made in 20 reputation systems. Some examples of different, novel approaches are provided from other systems as well, when they provide additional insight.

Our comparison covers the following reputation systems: Beta [32], eBay (Site 1), FuzzyTrust [71], PeerTrust [81], REGRET [66], Tran's [76], Travos [54], Trunits [36], TrustGuard [72], Credence [78], EigenTrust [35], NICE [68], PET [44], PowerTrust [87], P2PRep [5], Scrivener [52], SuperTrust [17], TrustMe [70], Buchegger's [8], and Li's [42].

These systems represent broadly three application areas: transactions in electronic marketplaces (Beta, eBay, FuzzyTrust, PeerTrust, REGRET, Tran's, Travos, Trunits and TrustGuard), collaboration in peer-to-peer networks (Credence, EigenTrust, NICE, PET, PowerTrust, P2PRep, Scrivener, SuperTrust and TrustMe), and routing in mobile ad hoc networks (Buchegger's and Li's). The systems cover a wide range of research areas of reputation systems; eBay is an implemented commercial reputation system included here for comparison.

4 Decomposition of Reputation Systems

In this section, we present our three-dimensional decomposition of reputation systems, in order to better understand the context of the requirements we proposed in the previous section, and the attacks and defenses in the next section. Although there are different reputation models and algorithms for a wide range of situations, the data used and the high-level output of reputation systems are not that different [38]. The overall reputation management process can be divided into three dimensions: 1) *dissemination*, which contains the communication of reputation information between participants in the system; 2) *calculation*, which takes the information collected by the first dimension as input and produces a decision as its output; and 3) *evolution*, which encompasses the updating of reputation information after an interaction occurs. The evolution dimension is often omitted in the existing literature. However, this dimension provides the important feedback loop that allows reputation systems to learn from new experiences, and is therefore also open to some of the most important weaknesses for reputation systems, such as issues with the fairness and the up-to-dateness of reputation information.

4.1 Dissemination

The dissemination of reputation information within the reputation system allows participants to learn from the experiences of others. While first-hand experiences are most reliable [44, 66, 67], they are not always available at the time of a decision. To reflect the possibility that shared information can be inaccurate, third-party information is typically given less weight in decisions [54]. Despite possible inaccuracies, information sharing allows misbehaviour to be punished by everyone in the system rather than just by the victim.

The dissemination process begins either when a user receives a request for recommendations (the "pull" approach, in e.g. HTrust [10]), or from an internal trigger to actively send out new information (the "push" approach, in e.g. eBay [55]). The recommending user then takes its local experiences and either preprocesses them into an overall opinion or prepares to send them out as single experiences or ratings, according to the defined format of recommendations in the system. It then follows the reputation system protocol to send the recommendation to the actor in charge of aggregating the results. Aggregation can be centrally performed by the reputation system or distributedly, by each actor making a decision [12]. In addition, mediators may be involved to forward recommendations in the network [65].

The information format contains two central variables: the metric used, and the format the disseminated recommendation takes. The dissemination protocol, in turn, is affected by the structure of the underlying network of users, and a possible overlay network structuring the peer-to-peer environment. Table 1 summarizes these aspects.

Table 1: Information dissemination dimension of selected reputation systems

System	Metric	Structure	Overlay	Recommendation
Beta [32]	continuous	centralized	none	counters
eBay [55]	discrete	centralized	none	rating
FuzzyTrust [71]	continuous	decentralized	DHT	rating
PeerTrust [81]	continuous	decentralized	DHT	rating
REGRET [66]	continuous	decentralized	none	aggregated
Tran's [76]	continuous	decentralized	none	unspecified
Travos [54]	continuous	decentralized	none	counters
Trunits [36]	N/A	centralized	none	unspecified
TrustGuard [72]	continuous	decentralized	none	unspecified
Credence [78]	discrete	decentralized	none	rating
EigenTrust [35]	continuous	decentralized	DHT	aggregated
NICE [68]	continuous	decentralized	none	unspecified
PET [44]	continuous	decentralized	none	aggregated
PowerTrust [87]	continuous	decentralized	DHT	aggregated
P2PRep [5]	continuous	decentralized	none	aggregated
Scrivener [52]	discrete	decentralized	DHT	unspecified
SuperTrust [17]	N/A	hybrid	none	unspecified
TrustMe [70]	N/A	centralized	none	unspecified
Buchegger's [8]	continuous	decentralized	none	aggregated
Li's [42]	continuous	decentralized	none	aggregated

The reputation metric sets the scale for expressing reputation. Many systems, including REGRET [66] and PeerTrust [81], use a continuous reputation metric, while some use a discrete metric (eBay [55], Scrivener [52], and Credence [78]). While a discrete metric is easy to understand for users, continuous variables are algorithmically simpler to handle.

The structure of many cited systems is decentralized, where all nodes make decisions on their own. For the handful of systems we categorize as centralized solutions, some services are provided by a central server. TrustMe [70], for example, relies on a central server to assign unforgeable identities, but reputation information dissemination is distributed. The hybrid approach of SuperTrust [17] includes a network of supernodes for storing ratings.

An overlay network helps manage the challenges with distributed information storage and dissemination. While most peer-to-peer systems on the Internet are unstructured [86, 87], several of the surveyed systems apply a distributed hash table (DHT) overlay (see e.g. [73]) to organize information stored by the peers.

The format of the recommendation can be single-transaction ratings or some form of aggregated value; there is a tradeoff between transparency and communication efficiency here. A set of ratings expressed as counters of satisfied and unsatisfied interactions (Beta [32] and Travos [54]) forms a special case of aggregation, which loses timing information. Other forms of aggregation include weighted averages of ratings (Regret [66], P2PRep [5]). Some systems, such as NICE [68], use generic opinions which have no direct relationship to transactions, or do not specify a standard recommendation format. Finally, while most systems use only their own first-hand experiences to produce recommendations, PET [44] uses an aggregated value of both first-hand and indirect information as its recommendation. When only first-hand information is used in recommendations, there is no risk of multiplying the input of a single experience by repetition [34].

Several systems specify a detailed dissemination protocol that aims to protect a specific desirable aspect in the system. Anonymity is a particular goal reflected in the protocols of TrustMe [70], SuperTrust [17] and P2PRep [5]. As reputation systems are a privacy tradeoff to begin with, sustaining some anonymity and user privacy are considered specialized targets for protection within the surveyed systems.

Dissemination is vulnerable to outside attacks. These can be mounted on the message level to prevent the propagation of experiences unfavorable to the attackers. Further attacks can be aimed to eliminate the negative information after dissemination by invading the storage nodes. We will give some specific examples of these possibilities in Sections 5.1.1 and 5.1.2.

4.2 Calculation

The reputation calculation dimension focuses on the decision-making support provided by reputation systems. The information collected from the first dimension is brought together with first-hand experiences to form a decision on

e.g. whether to engage in a transaction with the given actor. The calculation leading to the decision is made either by the interested user, a centralized entity, or by all nodes, as in the case of EigenTrust [37].

The most central aspect of the calculation dimension is the algorithmic approach taken. In addition, specific aspects of the calculation vary between systems: whether both positive and negative trust (distrust) are modelled, whether confidence in the resulting trust value or decision is modelled, whether the context of the trust decision is considered, and whether the credibility of third-party recommendations is analyzed. These aspects are summarized in Table 2.

Table 2: Reputation calculation dimension of selected reputation systems

System	Approach	Trust/Distrust	Confidence	Context	Rec. credibility
Beta [32]	probabilistic	both	none	none	content
eBay [55]	counting	both	none	none	N/A
FuzzyTrust [71]	fuzzy	no distrust	none	none	content
PeerTrust [81]	counting	no distrust	none	value	content
REGRET [66]	fuzzy, counting	both	knowledge, variability	value/quality/time	content
Tran's [76]	other	both	none	value/quality	N/A
Travos [54]	probabilistic	both	statistical analysis	none	content
Trunits [36]	other	N/A	none	value	N/A
TrustGuard [72]	counting	N/A	none	none	content
Credence [78]	flow	both	none	none	source
EigenTrust [35]	flow	both	none	none	content
NICE [68]	flow	both	none	none	content
PET [44]	counting	both	none	none	N/A
PowerTrust [87]	flow	no distrust	none	none	content
P2PRep [5]	fuzzy, counting	both	none	none	source
Scrivener [52]	other	N/A	none	none	content
SuperTrust [17]	counting	N/A	none	none	N/A
TrustMe [70]	N/A	N/A	none	none	N/A
Buchegger's [8]	probabilistic	both	none	none	content
Li's [42]	probabilistic	both	knowledge, variability	none	content

We present five categories of the algorithmic approaches in the literature.

Counting approach: We classify additive methods into this approach. Computing the reputation as the summation or average of all ratings is widely used in online reputation systems [33]. Among these systems, the simplest form of computing reputation is perhaps counting the positive and negative ratings, like eBay [55]. PET [44] follows this approach, but derives peer trustworthiness from a combination of long-term reputation evaluation and short-term risk evaluation. PeerTrust [81] accumulates the normalized amount of satisfactory experiences, weighted by their credibility. TrustGuard [72] calculates a weighted sum of current reputation, an aggregate of past reputation values, and their fluctuations.

Probabilistic approach: All systems in this category are based on the Beta probability density function. The Beta function is very suitable for processing binary outcomes, and it is used by the Beta reputation system [32], Travos [54], Buchegger's system [8], and Li's system [42] to evaluate reputation from a set of positive and negative experiences. Dirichlet functions have been proposed as a way to extend the basic binary division to a larger, discrete outcome scale [31, 56].

Fuzzy approach: Systems in this category apply fuzzy logic [85] to be able to express and reason about uncertainty in reputation information. REGRET [66] uses fuzzy rules when measuring recommendation credibility and neighborhood reputation. FuzzyTrust [71] defines fuzzy rules for computing local trust scores for the buyer and seller, and a global reputation.

Flow approach: This category includes systems that compute reputation based on the flow of transitive trust. EigenTrust [35] aggregates global trust scores by recursively calculating the left principal eigenvector of the matrix representing the local trust values, which are sums of positive and negative ratings. It assumes the existence of pre-trusted nodes to speed up the convergence of the computation, and to help resist malicious nodes. PowerTrust's [87] approach is quite similar to that of EigenTrust, but it uses a Bayesian method to generate the local trust values. Credence [78] calculates an estimate of correlation between any two connected nodes, also in a way similar to EigenTrust. Trust can also be propagated through looped or arbitrarily long chains in trust graphs. Of the reviewed systems, NICE [68] follows the chain approach. Guha et al. [24] discuss different forms of atomic trust propagation, including the propagation of distrust. Wang et al. [79, 80] define two operators drawn from path algebra [58] to deal with the trust propagation and aggregation along different chains. Mui et al. [50] define another widely cited computational model using chains.

Other approaches: Tran and Cohen [76] only consider direct experiences and employ reinforcement learning. Scrivener [52] only considers pairwise exchanges of content between overlay participants. Further, several systems utilize a map of social relations between peers to help decide their trustworthiness [22, 82]. This seems to be a promising approach in environments where this kind of social information is readily available, particularly considering that many attacks towards reputation systems have social origins [19].

Trust/distrust: Distrust is at least as important as trust [24], although it cannot be propagated the same way as trust: the enemy of your enemy is not necessarily a friend [74]. A few algorithms do not represent distrust information at all [71, 81, 87], while some use the number of satisfactory transactions minus the number of unsatisfactory transactions as a trust value [35]. NICE uses distrust information to help detect malicious users or collusion [68].

Confidence: Confidence measures how strongly we believe in the correctness of the trust/distrust opinion we have calculated. Sufficient and credible reputation information is needed to ensure confidence in the produced estimate of future behaviour, and the following decision. Flexible multilevel calculation, as first proposed in REGRET [66], can be applied when specific types of information are not sufficiently available. For example, if there are insufficient direct experiences or recommendations, the reputation of the target node can be deduced from that of its neighbors in a social network graph. Two more models (Travos [54], Li's [42]) consider the confidence of trust values. Confidence can increase with more knowledge [54], as well as less variability [66, 80], indicating an agreement among different sources. Users can thus try to gain more knowledge in response to disagreeing sources in order to reach the desired confidence.

Context: Context means the necessary information needed when making decisions. In the case of reputation systems, it includes the value, delivery time, and other qualities of the service claimed by the provider, as well as the specific type of the service. It is important and necessary to include the context information at the point of making decisions. Usually, a reputation system only deals with a limited set of context information. For example, PeerTrust [81] and Trunits [36] model the value of the transaction. In Tran's [76], sellers can adjust the quality and price of goods to maximize profits, and buyers may have different preferences over the goods. Similarly to Tran's, Roozmand et al. [59] follow the same scheme but consider one more facet of context information, the delivery time. REGRET [66] also divides reputation into three facets: price, delivery time, and quality.

Recommendation credibility: Credibility measures how strongly we believe the recommendation to be true. This belief can be divided into two parts, based on the source or the content of recommendation. If recommendations are only collected from a selected subset of users, or if information about the source is involved in the credibility analysis, it becomes important to verify the source as well. Approaches to verify the source of recommendation include digital signatures (e.g. Credence [78]) and random confirmation checks (e.g. P2PRep [5]). As for the content of recommendation, usually some weighting techniques are used to weaken the effect of potential inaccurate information. We will discuss them in the next section. They are also marked in the fifth column of Table 2.

Within the calculation dimension, we observe that the situation where the transaction takes place and the credibility of shared information are two important issues for protection and could be targeted by attackers. Attacks and defenses are given in Section 5.2.1 and 5.2.2. Source verification is further discussed in Section 5.1.1.

4.3 Evolution

Reputation evolution captures how the system learns from new information gained through interactions. First and foremost, new first-hand experiences are encoded in local reputation information. Some of the reputation information is also disseminated, and the arriving new information again causes changes in the reputation values. In addition, if recommender trustworthiness is tracked separately, its updates also fall under the evolution dimension.

Much of the previous work is surprisingly static, with little explicit attention given to reputation evolution [25]. There are three reasons for a serious consideration of the updating of reputation. First, reputation systems help decision making in an open environment, and should therefore be able to cope with changes in the environment and actors [11]. Second, as mentioned above, attackers could target the fairness and the up-to-dateness of reputation information. In addition to detecting direct misbehaviour, reputation evolution information can also help to detect malicious recommenders. HTrust [10], for example, detects dishonest recommenders by the frequency of opinion conflicts. Third, an actor's ability and willingness to provide a good service or relevant recommendations may change over time. As a result, both service reputation and recommender reputation should be tracked.

Many systems consider this evolution dimension only implicitly, i.e. discussing some aspects of it while not explicitly modelling the full updating mechanism, or treating it in an ad hoc manner. Various aspects of reputation evolution brought up in the literature include: 1) the updating of local knowledge based on new input, 2) the triggering of global reputation updates (i.e. when to activate the dissemination process) based on new information, and 3) detecting and reacting to behaviour changes. The definition of events that produce new experiences or ratings also often falls outside the scope of the reputation system specification; in eBay, users manually input their personal opinions into the system, but in many peer-to-peer environments, for example, automated detection of successful transactions is a more likely approach.

Local knowledge is updated through new local experiences as well as new recommendations. Tran and Cohen's reinforcement learning [76] is an incrementally learning algorithm. They also give update formulae for the expected product value and reputation rating. When using the approximate computation scheme, PeerTrust [81] caches the newly computed trust value. Buchegger's system [8] has two sources to update the reputation, direct experience and compatible indirect experience. Only recommendations that pass a deviation test are deemed to be compatible.

Recommendation credibility evolution, which takes care of how the credibility of recommenders should be updated based on the current recommendation, is not separately discussed by any of the surveyed systems. An exception is Buchegger's system [8], whose deviation test can update the recommender credibility in an ad hoc way. In contrast, two functions to maintain information about trustworthiness of an agent both as a service provider and a recommender are used by HTrust [10]. The maintained information is also used to detect dishonest recommenders. Hang et al. [25] propose a "max-certainty" update mechanism, where confidence about the trustworthiness of an agent which is held by recommenders is taken into account, and recommendations are compared with the direct experiences to help evolve the trust value of recommenders.

The trigger for global reputation updates, if specified, is typically the end of a transaction. Some systems (e.g. NICE [68] and Credence [78]) generate receipts to record direct trust between peers after each transaction. These receipts can still disseminate across the system even when the original recorders are not online. SuperTrust [17] also triggers dissemination of ratings to supernodes at the end of each interaction. PowerTrust [87] dynamically updates the calculated global reputation, especially that of power nodes, in order to identify new power nodes. In Trunits [36], a dishonest seller will lose the reputation units it has committed for a transaction, while an honest seller regains them and an additional reward.

More emphasis is needed on developing updating models and algorithms for reputation systems, especially compared with the extensive efforts in the reputation calculation dimension. The issues of detecting and reacting to behaviour changes and other detailed defense techniques for this dimension are discussed in Section 5.2.3.

5 Vulnerabilities and Correlative Attacks and Defenses

In this section, we demonstrate what kinds of vulnerabilities are created when one or more of the security and robustness requirements presented in Section 3 are not met. These vulnerabilities can be exploited in various ways to attack the system and its users. We also present different countermeasures proposed for the attacks. While some measures are very specialized, others represent good practices in reputation system design, and are collected into a set of guidelines for building secure and robust reputation systems in Section 7.

As our requirements, we also divide the related vulnerabilities into two categories: system-based vulnerabilities that relate to the foundation and environment of reputation systems, and metric-based vulnerabilities that tie to the selected reputation metric and its updates.

5.1 System-Based Vulnerabilities

5.1.1 Message Vulnerability

The message vulnerability category threatens the aspects identified in the dissemination dimension through attack opportunities directed at messages in transit. We have introduced the message authenticity, integrity and confidentiality requirement to address this vulnerability.

Attacks: An eavesdropper threatens message confidentiality only, while an active attacker can intercept messages and do just about anything to them in the open Internet infrastructure: it can stop the message from ever reaching its destination (e.g. to stop negative recommendations about itself), replace it with a message of its own (e.g. a fabricated positive recommendation), modify the original message before allowing it to continue, or simply copy it and replay the same message multiple times, in order to multiply the reputation effect of a single successful transaction.

Defenses: Attacks mounted on the network level, through e.g. spoofing IP addresses and subverting routers, cannot be fully defended against on the application level. However, some attacks are mounted on the application level by e.g. mediators in the reputation network. The surveyed systems apply cryptography, encryption and signatures, to protect message confidentiality, integrity and authenticity. In addition to signatures, some systems aim to protect message authenticity through random confirmation checks; this mostly ensures that assigned mediators in the network have not modified the messages.

PeerTrust [81] use two layers to ensure the security and integrity of data. The top layer is based on public key cryptography, where each peer of the system has a public and private key pair, while the bottom layer focuses on data availability. SuperTrust [17] adopts an additive encryption function which allows users to process the encrypted messages without opening them. This function helps preserve the sources' privacy and anonymity through the distribution and processing of trust ratings. TrustMe [70] also uses public key cryptography, but with separate key

pairs for transactions and reputation reporting. In P2PRep [5], the query node sends its public key to its peers, so that they can reply with an encrypted message. After that, the query node also chooses some peers to confirm their votes. Credence [78] battles fake votes by issuing digital certificates to the participants. To defend against the replay attack, TrustGuard [72] binds feedback to a specific transaction through a transaction proof which allows it to detect fake transaction reports.

Discussion: PKI-based schemes are commonly used in the surveyed systems to protect messages in transit. Typically, a requester sends his public key along with the request message, and the service provider encrypts the reply message with the requester's public key and signs it with its own private key. The service provider sends the reply message back together with its public key. However, these schemes require underlying infrastructure and processing power, and are therefore not necessarily applicable in e.g. wireless sensor networks [47] and mobile ad hoc networks.

5.1.2 Node Vulnerability

This vulnerability category concerns data storage and processing in the nodes. This is a particularly severe vulnerability for distributed storage; for example, the DHT overlay used by several reputation system proposals is not designed to be secure against malicious actors [14]. We have introduced the tamper-proofing and availability of stored reputation information requirement to address this vulnerability.

Attacks: We consider two kinds of attacks that strictly relate to reputation systems here: attacks against reputation data storage, and attacks against the node function in a Denial of Service (DoS). In addition to being a target of an attack, a node can misbehave itself. It can do this due to being malicious, or due to having been infiltrated by an attacker, as is usually the case with nodes controlled through malicious botnets. It may well be impossible for a third party to tell these two apart, however, and from an automation point of view it is simplest to react to all malicious behaviour in the same way.

Defenses: Protections to the message vulnerability, such as cryptography schemes and digital signatures, also protect against information tampering within the node. Denial of service is mostly protected against through redundancy, which can also provide protection for reputation data storage. Centralized systems provide a single attractive target for attack to bring the entire system down, which is why they should specifically be protected with redundancy. Distributed systems have their data and resources more evenly spread out, but they are also vulnerable to targeted attacks through the weakest links in the peer network [14, 16]. SuperTrust [17] works on a K-redundant superpeer network. EigenTrust [35] assigns a set of peers to compute one peer's trust value and then takes a majority vote among them, with the assumption that dishonest nodes are a minority. PeerTrust [81] also uses data replication at its bottom layer. In TrustMe [70], trust values are stored in several trust-holding agent peers randomly chosen, and a majority vote is taken to select the final value. Scrivener [52] also utilizes redundancy to check the validity of certain claims.

Discussion: Redundancy is essential as malicious nodes might discard data in local storage. Pre-trusted nodes, if they exist, could be used to store reputation data. In this case, redundancy is still necessary as pre-trusted nodes are likely to become the targets of a DoS attack.

5.1.3 Identity Management Vulnerability

The identity management vulnerability results from the openness of reputation systems, as nodes can enter and leave the system more or less freely. While this is a desired property of most reputation systems, it introduces identity management issues that need to be solved. We have introduced the leverage-balanced identity management requirement to address this vulnerability.

Attacks: In a re-entry attack, a malicious node discards its old disreputable identity and re-enters the system with a new identity. For a Sybil attack [18], a malicious node enters the system with multiple identities in order to increase its leverage in the community.

Defenses: This vulnerability is extremely difficult to handle in the absence of a central authority who can assign identities in the network based on e.g. the actual identity of the user. Without such a central authority to limit the number of identities a user can assume, a common solution of the surveyed systems is to introduce a cost for generating a new identity. The cost is not necessarily monetary, but it can consist of an investment in time or resources. For example, heavy computation might be required before granting a new identity, such as in Credence [78] does. Another solution is to give some privileges to long-term identities, as proposed in NICE [68]. This policy encourages users to maintain a long-term identity, since a new identity needs to pay more for the same service or has limitations in the amount of trade it is allowed to perform.

Discussion: Cheng and Friedman theoretically prove that in the absence of an identity management infrastructure, global reputation values calculated from an open set of participants can always be subverted by the Sybil attack, while subjective reputation calculated using a trust flow approach appears to be more resistant [13]. Social networks provide a basis for a promising defense against Sybils: Yu et al. [83, 84] detect Sybils on a social graph based on the insight that the removal of a small set of edges from Sybils will completely disconnect the Sybil identities. Mislove et

al. [49] only assign reputation to pairwise relationships based on the assumption that users in a social network do not create relationships arbitrarily. As a result, the effects of Sybils are limited as it is hard for multiple identities to connect to a single user.

5.2 Metric-Based Vulnerabilities

5.2.1 Lack of Context Awareness

Context awareness means including information about the situation of the decision, such as the value of a transaction, in the computational model. When this information is missing, all transactions are treated equally, which causes problems when they are not truly interchangeable. We have introduced the requirement for context-aware decision making to address this vulnerability.

Attacks: In a value imbalance attack, a seller honestly executes small sales and cheats on large ones [37]. In a reputation type exploitation, a service provider who is e.g. good at reporting weather might behave poorly on booking tickets. This is, in essence, a generalization of the attack where a node provides good service but unfair recommendations [21]; we discuss this specific issue further in the next section.

Defenses: PeerTrust [81] considers transaction context, which can include the value of the transaction. Trunits [36] deals with the value imbalance problem directly by requiring more trunits to be stored at a trusted third party (and lost, in case of misbehaviour) for more valuable transactions. Tran and Cohen's [76] reinforcement learning takes into account the prices of goods in calculating the expected value of the transaction, but it mainly focuses on the adaption of prices. REGRET [66] considers reputation to have multiple facets, which means that e.g. shipping time and goods quality can be traced separately. For the reputation type exploitation problem, a provider–service–value tuple can be used instead of just the provider–value one; different proposals have been made on explicitly encoding the meaning of a rating for a specific type of application environment [3, 69]. In addition, some systems [45, 66] adopt the ontology techniques from Semantic Web to compute the similarity between contexts in order to avoid issues with the data sparsity that can follow from distinguishing between too many different contexts.

Discussion: Reputation is context-dependent and context itself is a multi-facet concept [28, 45, 51]. It is hard to formalize the context and facets of different applications in a general way. Several existing proposals already incorporate the value of the transaction in the reputation model. Tracking reputation separately for different tasks seems to be a reasonable approach, but must be combined with means to translate reputation information between semantically close contexts in order to ensure that there is at least enough information for a low-confidence decision even if the contexts between earlier experiences and current decision are slightly different. In earlier work, Ruohomaa and Kutvonen have proposed a compromise in the form of tracking reputation based on a set of different assets [63] rather than keeping track of all the activities that may influence them differently; however, this proposal is itself based in an environment where reputation is collected primarily on specific services rather than on the enterprises providing them.

5.2.2 Reputation Fabricability

One central goal of reputation systems is to generate social pressure to behave well, as misbehaviour threatens to lower the node's reputation and reduces its attractiveness as a service provider [57]. However, measuring reputation also creates the motivation and possibility for reputation fabrication. We have introduced the recommendation credibility evaluation requirement to address this vulnerability.

Attacks: On a high level, reputation attacks consist of either defamation, i.e. slandering honest nodes, or whitewashing, i.e. undeservedly boosting the reputation of malicious nodes. Specialized attacks include e.g. collusion, where multiple nodes cooperate to produce the above effects, and moles [21], who are honest in actual transactions but provide dishonest ratings when it benefits them.

Defenses: Many systems have studied the problem of unfair ratings. For example, EigenTrust [35] experiments on the moles threat, and malicious collusion to mount reputation attacks is studied in NICE [68], PeerTrust [81], and PowerTrust [87]. As a defense, we focus on the credibility of recommendations. Recommendation credibility computation can be done based on 1) the actual content of the recommendation and how well it matches expectations set by other information (e.g. the similarity-based method), or 2) information about its source, such as the source's reputation as a service provider (reputation-based) in general or its track record as a recommender specifically (second-order reputation-based) [33, 65]. The similarity-based credibility measure calculates the similarity of two nodes by comparing the opinions that they hold towards any other nodes that have had interactions with both of them. The reputation-based credibility measure assumes that a node with good behaviour always provides honest recommendations and vice versa. The third credibility measure, based on second-order reputation, models the recommender's track record separately. Beta [32] and Li's [42] systems follow subjective logic [29] (also inherently reputation-based) and model uncertainty explicitly.

Systems that only use direct experiences are not threatened by these attacks, but instead suffer from information sparsity. Nearly all other systems using recommendations make a recommendation credibility analysis. Also, if

negative feedbacks are not used, there is no problem of slander, but similarly there is even less defense against whitewashing.

Discussion: Reputation-based recommendation credibility is vulnerable to moles, as moles perform inconsistently at providing services and recommendations. Selecting recommenders instead based on how well their past recommendations match local experiences gives better results. Experimental results of TrustGuard show that similarity-based recommendation credibility outperforms the reputation-based in situations where malicious nodes take a large share and when the malicious nodes form collusions [72]. We conclude that keeping a track record of recommender behaviour is the best solution of the three; it presents a system of second-order punishment (i.e. punishing those who unfairly punish misbehaviour), which has been shown to be vital in maintaining social control within large communities [20].

5.2.3 Reputation Incongruity

Reputation systems are built on the assumption that a history of past behaviour can be used to predict future behaviour. The reputation incongruity vulnerability results from this assumption being violated. The central problem revolves around reacting to changes. We have introduced the incongruity management in reputation evolution requirement to address this vulnerability.

Attacks: In the basic attack, nodes use old good reputation built by honest behaviour to cheat for profits. An oscillating node behaves honestly for a while to attain a good reputation, then cheats until its reputation is lost, and repeats this process [72]. Discrimination can also mislead reputation predictions: a node can provide good service to specific nodes and bad service to the rest, or vice versa. For example in systems such as EigenTrust [35] and PowerTrust [87], the node may provide good service to the pre-trusted nodes only, or only to the ones with a high reputation in systems using reputation-based recommendation credibility. Finally, as there always exists a time lag between the service provision time and the time of the corresponding experience report being disseminated [30], there is a reputation lag during which a node can cheat for a while before anyone else is notified. This problem is particularly notable in transactions where goods must be delivered over long distance, and honest participation can only be confirmed as the goods arrive.

Defenses: To ensure accurate predictions of future behaviour based on reputation, it is important to detect and react to any changes in behaviour. For this purpose, it is widely believed that good reputation should be slowly gained but quickly lost. For example PET [44] follows this principle. Some systems (Beta [32], Buchegger's [8] which also gives a redemption opportunity for misbehaviour, and P2PRep [5]) discount old information in favour of new, to acknowledge the importance of current behaviour over the past. TrustGuard [72] explicitly models reputation fluctuations to counter the oscillation attack. For the discrimination attack, similarity-based recommendation credibility is considerably more resistant than reputation-based recommendation credibility due to its subjectiveness, while upkeeping a local recommender credibility score is even better. To solve the problem of discrimination, Dellarocas [15] has proposed to conceal the identities of different service requesters. The reputation lag problem is impossible to solve completely because of the unavoidable time lag in transmission of messages and goods. Some efforts have been made against the lag of goods transmission, however: in Trunits [36], a portion of the seller's reputation is bound to each transaction and kept in a trusted third party, and is only released once the buyer approves the delivered goods. A similar scheme using a trusted third-party arbitrator that stores money instead of reputation has been proposed e.g. by Li and Martin [43]; this incurs a requirement that the electronic marketplace be centrally controlled by a party that all participants can trust with their money.

Discussion: To alleviate the problem of reputation incongruity, it is necessary to integrate some form of a time dimension and updating mechanism into the reputation model. Ruohomaa et al. have proposed reputation epochs as a mechanism to better react to behaviour changes specifically [61, 63]. An epoch is a period of consistent behaviour sequences (e.g. all positive experiences), and it can be optimized to different applications by adjusting the epoch detection algorithm and the weighting of the most recent epochs.

6 Environmental Analysis

In this section, we discuss the different measures proposed in the compared systems within the context of their application areas, be it electronic marketplaces, peer-to-peer networks, or mobile ad hoc networks. These are the three most widely used areas of reputation systems. While different application areas have different goals, they are also sensitive to different vulnerabilities.

We summarize the measures each system takes towards meeting our robustness and security requirements in Table 3. The numbers in the bracket and the following notes connect the measures to the corresponding requirements.

We first focus on the nine reputation systems for *electronic marketplaces*. Beta [32] and Trunits [36] are centralized reputation systems designed for the e-Commerce environment. Beta builds on the Beta probability density function, evaluates recommendation credibility based on subjective logic, and discounts old ratings. The basic idea in Trunits

is quite different. Trunits treats reputation as money. The value of transactions is reasonably modelled into the system through a reputation mortgage which is proportional to the goods value.

Table 3: Protection for requirements of selected reputation systems

System	Protections for the six requirements
Beta [32]	(5) subjective logic recommendation credibility, (6) old ratings discounting
eBay [55]	(1,2,3,4) covered fully or in part through a centralized server
FuzzyTrust [71]	(5) reputation-based recommendation credibility
PeerTrust [81]	(1) public key cryptography, (2) data replication, (4) transaction value considered, (5) reputation-based/similarity-based recommendation credibility
REGRET [66]	(4) multi-faceted reputation, (5) reputation and social relation based recommendation credibility, (6) old ratings discounting
Tran's [76]	(4) transaction value considered, (5) only direct experiences used
Travos [54]	(5) recommendation record based recommendation credibility
Trunits [36]	(4) reputation mortgage proportional to transaction value, (5) only direct experiences used
TrustGuard [72]	(1) fake transaction report detection, (5) similarity-based recommendation credibility, (6) reputation oscillation modeling
Credence [78]	(1) digital certificate, (3) expensive computation requirement, (5) similarity-based recommendation credibility
EigenTrust [35]	(2) data and computation replication, (5) reputation-based recommendation credibility
NICE [68]	(1) digital certificate, (3) pricing privilege, (5) only direct experiences used
PET [44]	(6) slow gain and quick loss of reputation
PowerTrust [87]	(5) reputation-based recommendation credibility
P2PRep [5]	(1) public key cryptography, random confirmation, (6) old ratings discounting
Scrivener [52]	(2) data replication, (5) only direct experiences used
SuperTrust [17]	(1) public key cryptography, (2) structural redundancy
TrustMe [70]	(1) public key cryptography
Buchegger's [8]	(5) reputation-based recommendation credibility and deviation test, (6) old reputation discounting
Li's [42]	(5) subjective logic recommendation credibility

FuzzyTrust [71] and PeerTrust [81] are two decentralized peer-to-peer e-Commerce reputation systems. From the view of robustness and security, PeerTrust provides protections for the message vulnerability by public key cryptography, for the node vulnerability by data replication, for the context vulnerability by considering the transaction value, and for the reputation credibility by applying a reputation-based or similarity-based measure. In contrast, FuzzyTrust only includes a reputation-based reputation credibility measure, and does not address the other requirements.

REGRET [66] and Tran's [76] are two decentralized systems proposed for multi-agent marketplaces. REGRET filters recommenders and aggregates recommended values based on a social network analysis. It also models multiple facets of reputation and discounts old transaction outcomes. Tran's system expects that a high reputation can boost the sales and allow higher prices. There are no recommendation credibility issues, as only direct experiences are used in the system.

Travos [54] aims to ensure good transactions in multi-agent system. It follows Beta [32] by employing the Beta probability density function. However, Travos uses a different method to evaluate the recommendation credibility, tracking the past recommendation behaviour of recommenders. TrustGuard [72] builds on PeerTrust, and provides reputation management for e-Commerce. It adds three components to deter oscillation attacks, fabricated recommendations on nonexistent transactions, and dishonest recommendations about real transactions, separately.

Among the many rigorous and complicated reputation models or systems in the electronic marketplace, eBay stands out as the only deployed system, and applies a reasonably simple algorithm. For message security, eBay uses SSL encryption when users log in, and saves some details of the transaction history for further examination to determine transaction context. All recommendations are stored and processed by a centralized server.

As a comparative study, we now discuss the robustness and security measures of reputation systems for cooperation in *peer-to-peer networks*. EigenTrust [35] is a widely studied and cited reputation system for file-sharing in peer-to-peer networks. It uses a distributed hash table (DHT) as a layout to store and compute reputation scores. Similar to EigenTrust, PowerTrust [87] also uses DHT and a flow-based reputation calculation approach. PowerTrust's main improvements to EigenTrust are enhanced performance and consideration of the power-law distribution of feedbacks.

Credence [78] is a peer-to-peer file-sharing reputation system deployed on Gnutella. It uses digital certificates to prevent fake votes and requires a central server or expensive computation to mitigate the problem of multiple

identities. As to recommendation credibility, Credence takes a similarity-based approach, based on an assumption that dishonest users tend to vote randomly while honest ones do not. PET [44] also aims to help cooperation in a peer-to-peer resource sharing setting. It applies a short-term risk evaluation to keep track of the recent performance of a specific user. This short-term evaluation can mitigate the reputation incongruity vulnerability.

Scrivener [52] aims to discourage freeloading in content distribution systems. It tries to find a credible path to the desired content for the requester, and avoids the recommendation credibility problem by only using first-hand experiences. NICE [68] is another reputation system based on path finding to boost peer-to-peer cooperations. It specifies limitations to what new identities are allowed to do, and gives pricing advantages to long-lived identities to discourage re-entries to the system.

P2PRep [5], SuperTrust [17], and TrustMe [70] are three reputation management frameworks that focus on the low-level infrastructure for peer-to-peer systems. P2PRep proposes a protocol using public key cryptography and random confirmation to protect against message vulnerability. Old reputation information is also discounted in P2PRep. SuperTrust is designed for superpeer networks, and it protects against message vulnerability and node vulnerability through encryption techniques and structural redundancy. TrustMe also uses public key cryptography and hides the rating node's identity to reduce reciprocity. There is also a central server in TrustMe to assign identities to newcomers.

Reputation systems in *mobile ad hoc networks* are different from the above systems due to the lack of infrastructure and the limited resources [45]. For example, multiple identities are difficult to deter as there is no central server, and PKI algorithms may be too expensive for their worth, as these algorithms often are computationally intense.

Among the surveyed systems, Buchegger's system [8] and Li's system [42] are two examples for mobile ad hoc networks. Buchegger's system uses a deviation test to evaluate the recommendation credibility, and a reputation discounting mechanism to protect against reputation incongruity. Li's system extends the idea of Beta distribution and subjective logic of the Beta system [32], to apply it in the mobile environment.

Based on the above analysis, we now summarize the environmental properties and the corresponding measures of reputation systems based on the requirements we proposed in Section 3.

Nearly all the systems we surveyed in electronic marketplaces do not consider the first three robustness and security requirements, namely the *message authenticity, integrity and confidentiality requirement, tamper-proofing and availability of stored reputation information requirement, and leverage-balanced identity management requirement*. An explanation might be that e-Commerce environments often adopt a centralized structure. The central authority can help users handle the vulnerabilities of the underlying network, such as in eBay. In contrast, several systems in peer-to-peer environments consider these requirements because of its open, distributed, and anonymous nature. For example, encryption techniques, replications, and more centralized solutions such as superpeers or servers are applied to fulfil the requirements.

Mobile ad hoc networks are completely distributed and lack computational resources. As a result, reputation systems here seldom make an effort on these three requirements. However, such proposals do exist; further discussion on electronic signature solutions in mobile devices can be found in [60].

As to the *context-aware decision making requirement*, it is particularly important for reputation systems in electronic marketplaces to take into account the value of the goods. Four (PeerTrust [81], REGRET [66], Tran's [76], and Trunits [36]) of our surveyed systems incorporate goods value in the reputation metric, and they all target this environment. To improve user personalization and satisfaction, REGRET [66] and Tran's [76] also consider the issue of different reputation facets. Peer-to-peer and mobile environments might also incorporate the importance of the protected resource into reputation systems in the future.

For the *recommendation credibility evaluation requirement*, which is relevant for all systems using third-party recommendations, nearly all surveyed systems believe that recommendations should be weighted (in systems that focus on reputation calculation, e.g. PeerTrust [81]) or checked (in systems that focus on underlying infrastructure, e.g. P2PRep [5]). The latter approach alone does not protect against dishonesty, however.

Intuitively, the *incongruity management in reputation evolution requirement* should be considered in all areas equally, just like the previous requirement. After analyzing the systems, we find that although there is no particular preference in different areas, not many systems try to meet this requirement in contrast to the recommendation credibility requirement. The simple measures of giving more weight to more recent information or to negative experiences do not really solve the problem. Novel solutions are needed for this issue, such as a model for tracking behaviour changes over time specifically.

Besides the above requirements, some other differences are found. Negative reputation values reflect the fact that sellers should lose some reputation in society due to dishonest behaviour. As a result, usually, reputation can be either positive or negative in electronic marketplaces. In peer-to-peer resource sharing communities, however, peers are unlikely to accept a negative reputation, when they can create a new identity instead [5]. Rather, these systems

aim to simply promote good participants over others. Negative information is important to a mobile ad hoc network, however, as its main goal is to detect and isolate the misbehaved nodes.

Another difference concerns confidence. It is necessary for reputation systems in electronic marketplaces to model confidence, since every transaction directly relates to the profits of sellers and buyers. Of the three systems that support confidence, REGRET [66] and Travos [54] are two examples in electronic marketplaces. Li's system [42] for mobile ad hoc networks models uncertainty to reflect the lack of confidence. None of the systems in peer-to-peer networks address confidence, on the other hand.

7 Guidelines for Developing Robust and Secure Reputation Systems

In this section, we set out to fulfil the requirements set in Section 3, and present guidelines based on and extending the topical practices gleaned from our literature review. All system design should involve an analysis of the specific robustness and security needs of that system. However, we assert that in the case of reputation systems, particularly in the context of e-Commerce, our six requirements should always be fulfilled.

The first three requirements have a strong influence on the dissemination dimension of reputation systems, as described in Section 4. While there are ready solutions available for e.g. cryptographical services and the issues may therefore seem unimportant, applying the solutions may interfere with the reputation system if the requirements are not explicitly considered in its design. This is why we find that in order to attain credibility, a reputation system must address these "low-level" requirements as well.

Ensuring message authenticity, integrity and confidentiality is a task which naturally lends itself to being solved with a cryptography scheme. Typically asymmetric, public key cryptography is used, as it does not require a secure channel for distributing any shared secret keys.

For authenticity, public keys must be reliably bound to the identities of the actors within the reputation system. A public key infrastructure (PKI) is needed to ensure that the key actually belongs to the given communication partner, and not an impostor intercepting the communication. Centralized certificate authorities, such as Verisign (Site 2), can bind a key to e.g. the real-world identity of a service provider. Also, a centralized identity management system for an electronic marketplace can act as a certificate authority when it generates identities that only exist within the marketplace: public keys can then form the actual identity of an actor. The issues of PKI in different environments are further discussed in related work [9, 40, 41].

Once the key infrastructure is in place, ensuring integrity and confidentiality becomes relatively simple: signing all messages with the sender's secret key ensures that tampering is detected, and encrypting all messages with the recipient's public key ensures that only the recipient can read their contents. As such cryptography services are widely available in ready implementations, there is no need to invent new schemes from scratch unless the application area is very constrained on processing power. The main concern is whether there are issues in fitting signatures and encryption with the dissemination model, as for example anonymous recommendations will require special attention.

Tamper-proofing and ensuring the availability of stored reputation information is most directly addressed by storing the relevant reputation information in each decision-making node. Of all forms of tampering, information omission is most difficult to protect against with cryptography, and a third-party storage is in an excellent position to delete information that could be beneficial to its competitors or detrimental to its allies. Trusted third-party witnesses, i.e. notary services, form a potentially expensive but effective protection against omission, in addition to also enabling an audit trail useful for verifying specific experiences.

First-hand storage comes with a communication and storage cost. In addition, many reputation schemes will want to provide newcomers with some existing, shared information as well, and in some cases this requires extending trust towards the mediators caching this information. When shared storage is used, the application of a central server promises to solve most issues, but in return it must be trusted not to tamper with the information itself. Protection against denial of service attacks would then be mostly a matter of resource replication.

In the electronic marketplace, banks taking care of monetary traffic typically already form one type of trusted actor. It could be reasonable to assume that a similar service, building its business on trustworthiness, could be employed to handle reputation information bootstrapping for newcomers. It could handle the storage altogether, although at a greater risk. Currently, such solutions are provided in traditional marketplaces by e.g. accreditors such as the Better Business Bureau (Site 3), and credit ratings companies, such as Standards & Poor's (Site 4). Relying on a single commonly trusted actor forms a weakness in the system that may be fatal to e.g. actors who somehow threaten the business interests of the trusted reputation cache. Instead of promoting monoculture, therefore, even trusted centralized storage must have competitors to discourage misbehaviour.

When relying on a single trusted party is not plausible, a reasonable middle ground between distributed and centralized storage can be found in primarily storing all relevant information in the decision-making node, while using

a scheme for selecting some pre-trusted nodes for each newcomer to provide enough reputation information for bootstrapping. While this method results in different information stored in different nodes and is therefore not as elegant from a modeller's perspective, it is essentially a generalization of the single trusted third party. In this case, actors can choose for themselves which third party to trust. These could, in turn, be tied to out-of-band trust relationships that the new actors have themselves: besides the major bank-like actors we mentioned earlier, or government facilities, the actors' old business partners could act as their electronic marketplace equivalent of mentors. Ways of locating such trusted third parties have been discussed by e.g. Alcade [2].

When no pre-existing trust relationships are available, information must be stored in a fully distributed manner. At the moment, this approach carries issues that are not entirely solved. Distributed hash table (DHT) overlays are designed to provide distributed information storage and dissemination, but as noted in Section 5.1.2, the redundancy they apply is only secure against random failures, not a targeted attack. Further discussion on peer-to-peer storage can be found in e.g. [26].

Providing leverage-balanced identity management requires that the cost of creating multiple identities is in balance with the leverage gained with the new identities. In an e-Commerce setting, centralized identity management may be adopted in order to tie the marketplace identities to real-world entities and to, at least in theory, allow legal recourse if a transaction goes wrong. With this solution, creating multiple identities becomes sufficiently costly to not be an issue. Another option is to bind identities to other limited resources, such as phone numbers or, recursively, identities in other systems bound to limited resources or real-world identities.

If the above options are not plausible, we must consider whether additional leverage gained in the reputation system is worth real money. If it is, as it may well be in an electronic marketplace, simple computational puzzles are unlikely to sufficiently increase the price of each identity to balance for the gained leverage. In this case, a new identity must either cost money, or its leverage must be drastically reduced.

The leverage of multiple identities can be reduced by using a subjective, flow-based reputation metric rather than a symmetric or "global" reputation metric. In a flow-based metric, influence in the system is not determined by the number of votes you can give, but the number of actors who trust you and who, in turn, are trusted by other actors. An isolated clique of Sybils that has no trust relationship with honest nodes cannot influence decisions outside the clique. In summary, making multiple identities impossible is hardly necessary; they simply must cost enough or be too inefficient that e.g. a Sybil attack is no longer attractive.

Supporting context-aware decision making is another requirement where neither extreme, context-free decisions nor very deeply classified contexts, solve the actual needs of a reputation system. Some context is needed, but some generalizations are also necessary or the available information becomes too sparse to be useful. For e-Commerce, we propose two basic features to consider:

- For reputation information, provide a measure of the value (or utility) wagered in a transaction. Divide this into more dimensions as necessary, but separate the scales for how well expectations were fulfilled and what the value of the transaction was. Use the value of the transaction to give more weight to high-value transactions in reputation calculation.

- For decision-making, separate reputation earned from high-value transactions from that from low-value transactions, and use the appropriate category primarily for the decision, depending on the stake at hand. A lower-confidence decision can be produced with mismatched information, i.e. a decision about high-value transactions deduced from information about low-value transactions, but for this the system should also be able to take advantage of confidence information.

If factors such as competence for the task [77] must also be considered, they may be best to include in the decision separately. While separate reputation scores can reasonably be kept on a handful of orthogonal tasks, more detailed matching requirements may make information too sparse, and therefore require translation of reputation from one category to another. Again, if confidence in the resulting decision is represented separately, these kinds of translations between categories can be done when insufficient information would otherwise be available.

Evaluating recommendation credibility is the single most important method for catching misinformation from the recommenders. Lying about the behaviour of other nodes should always have a negative impact on reputation; only this kind of second-order punishment system allows communities to scale up in size. Due to this, basing credibility as a recommender only on a node's behaviour as a service provider is insufficient; it leaves the system vulnerable against moles.

In addition to analyzing the credibility of the recommendation source, the recommendation itself can be examined in its context. Recommendations differing from majority opinion can be signed off as outliers, for example; whether a single negative recommendation is the result of a dishonest recommender or the service provider behaving discriminatorily against that particular recommender, the usefulness of the recommendation is likely to be low for this specific decision-maker. On the other hand, recommendations similar to the evaluator's own experiences may be

readily accepted as supporting evidence. In fact, should the evaluator be subjected to discrimination itself, only recommenders similar to itself are of any use as information sources.

The details of how the "credibility reputation" of a recommender should be updated, and how to configure the combined credibility analysis, are decisions more dependent on the application context. We find that the former question merits further research, however, and are currently investigating contractually governed reputation systems based on objective, verifiable experiences [64].

Finally, *managing incongruities in reputation evolution* requires changing our viewpoint from information gathering to change detection. As a decision-making tool, a reputation system actually has two major tasks: sorting other actors into well-behaved and ill-behaved, and reacting promptly when this categorization turns out to be incorrect. Supporting both of these two goals with a single, flat reputation information model is a daunting task, and the reputation evolution aspect has been widely neglected.

Instead of simply waiting for gained positive reputation to be used up due to misbehaviour, reputation systems must actively aim to detect and react to incongruities and provide information about them to use in decisions. The reputation epoch mechanism we have described in earlier work by Ruohomaa et al. [61] is one flexible approach to this, and we will expand on it in future work. However, it is only one possible solution. The lessons learned from anomaly detection within computer network security should prove useful for this goal, as well as data mining research on novelty detection in time series (for an introduction, see e.g. [75]).

8 Conclusion

In this paper, we have identified and analyzed common requirements and vulnerabilities of reputation systems in different application areas. Through applying our decomposition framework to a set of selected reputation systems, we have drawn relevant attacks and defense mechanisms to shared vulnerabilities. Moreover, we have studied different application areas of reputation systems, and categorized the environmental characteristics in terms of the requirements. Finally, we have provided a set of guidelines to build robust and secure reputation systems.

Many attacks towards reputation systems have social origins, and may therefore be best addressed through solutions based on the same theme. As with the Sybil attack, social relationship modelling may prove to be better at solving many of our problems than any flat statistical analysis. The question then becomes how to detect these relationships, particularly in distributed systems without any global trust information available. This should provide an interesting avenue for future work.

Security and robustness require tradeoffs. Instead of aiming for producing impenetrable systems that may be entirely unusable, a balance must be sought between costs and gains. It is particularly useful to know and realistically model the cost and value of each attack from the attacker's perspective. The more valuable a good reputation is, the higher motivation there is to mount a reputation attack. Making an attack sufficiently costly that it becomes less attractive to an attacker is a simpler and often better solution than making the attack downright impossible. In the end, instead of addressing possible attacks one at a time, we should focus on addressing vulnerabilities, through identifying the security and robustness requirements of the application context and fulfilling them to our best ability.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 60973044, 60903024, 60736015, 60721002, 61073030), the National 973 Program of China (2009CB320702), the National 863 Program of China (2009AA01Z117), the National Special Program for Grand Science and Technology Challenges of China (2009ZX01043-001-06), and the "Climbing Program" of Jiangsu Province, China (BK2008017). Sini Ruohomaa works as a part of the CINCO group at the University of Helsinki. We thank the anonymous reviewers for their helpful comments and suggestions on improving this manuscript.

Websites List

Site 1: eBay--The World's Online Marketplace
http://www.ebay.com/

Site 2: VeriSign--Internet infrastructure services for the digital world, Security (SSL Certificates), Domain Name Services, DDOS Mitigation and Identity Protection
http://www.verisign.com/

Site 3: Better Business Bureau
http://www.bbb.org/

Site 4: Standard & Poor's website

http://www.standardandpoors.com

References

[1] B. T. Adler and L. de Alfaro, A content-driven reputation system for the Wikipedia, in Proc. of the 16th International Conference on World Wide Web (WWW'07). Banff, Alberta, Canada: ACM, May 2007, pp. 261–270.
[2] B. Alcade, Trusted third party, who are you? in Short Paper Proceedings of the Fourth IFIP WG11.11 International Conference on Trust Management (IFIPTM 2010), Morioka, Iwate, Japan, jun 2010, pp. 49–59. [Online]. Available: http://www.ifip-tm2010.org/lib/exe/fetch.php?media=shortpaper07.pdf
[3] R. Alnemr, S. Koenig, T. Eymann, and C. Meinel, Enabling usage control through reputation objects: a discussion on e-commerce and the internet of services environments, JTAER: Journal of Theoretical and Applied Electronic Commerce Research, vol. 5, no. 2, pp. 59–76, 2010.
[4] D. Alperovitch, P. Judge, and S. Krasser, Taxonomy of email reputation systems, in Proc. of the 1st International Workshop on Trust and Reputation Management in Massively Distributed Computing Systems (TRAM'07), in conjunction with IEEE ICDCS 2007. Toronto, Canada: IEEE Computer Society, June 2007.
[5] R. Aringhieri, E. Damiani, S. D. C. Di Vimercati, S. Paraboschi, and P. Samarati, Fuzzy techniques for trust and reputation management in anonymous peer-to-peer systems, Journal of the American Society for Information Science and Technology, vol. 57, no. 4, pp. 528–537, February 2006.
[6] D. Artz and Y. Gil, A survey of trust in computer science and the semantic web, Web Semantics: Science, Services and Agents on the World Wide Web, vol. 5, no. 2, pp. 58–71, 2007.
[7] K. Böhm, S. Etalle, J. Den Hartog, C. H¨ utter, S. Trabelsi, D. Trivellato, and N. Zannone, A flexible architecture for privacy-aware trust management, JTAER: Journal of Theoretical and Applied Electronic Commerce Research, vol. 5, no. 2, pp. 77–96, August 2010. [Online]. Available: http://dx.doi.org/10.4067/S0718-18762010000200006
[8] S. Buchegger and J.-Y. Le Boudec, A robust reputation system for mobile ad-hoc networks, KTH Royal Institute of Technology, Theoretical Computer Science Group, Techical Report, 2004.
[9] S. Capkun, L. Buttyán, and J.-P. Hubaux, Self-organized public-key management for mobile ad hoc networks, IEEE Transactions on Mobile Computing, no. 1, pp. 52–64, 2003.
[10] L. Capra, Engineering human trust in mobile system collaborations, in Proc. of the 12th ACM SIGSOFT twelfth international symposium on Foundations of software engineering. Newport Beach, CA, USA: ACM, October 2004, pp. 107–116.
[11] S. Casare and J. Sichman, Towards a functional ontology of reputation, in Proc. of The 4th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'05). The Netherlands: ACM, 2005, pp. 505–511. [Online]. Available: http://doi.acm.org/10.1145/1082473.1082550
[12] D. W. Chadwick, Operational models for reputation servers, in Proc. of the 3rd International Conference of Trust Management (iTrust'05). Paris, France: Springer-Verlag, May 2005, pp. 108–115.
[13] A. Cheng and E. Friedman, Sybilproof reputation mechanisms, in Proc. of the 2005 ACM SIGCOMM Workshop on Economics of Peer-to-Peer Systems. Philadelphia, Pennsylvania, USA: ACM, August 2005, pp. 128–132.
[14] S. Dahan and M. Sato, Survey of six myths and oversights about Distributed Hash Tables' security, in Proc. of the 1st International Workshop on Trust and Reputation Management in Massively Distributed Computing Systems (TRAM'07), in conjunction with IEEE ICDCS 2007. Toronto, Canada: IEEE Computer Society, June 2007. [Online]. Available: http://dx.doi.org/10.1109/ICDCSW.2007.77
[15] C. Dellarocas, Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior, in Proc. of the 2nd ACM Conference on Electronic Commerce. Minneapolis, Minnesota, USA: ACM, October 2000, pp. 150–157.
[16] P. Dewan and P. Dasgupta, Securing P2P networks using peer reputations: Is there a silver bullet?, in Proc. of the 2nd IEEE Consumer Communications and Networking Conference (CCNC'05), Las Vegas, Nevada, USA, January 2005, pp. 30–36. [Online]. Available: http://dx.doi.org/10.1109/CCNC.2005.1405139
[17] T. Dimitriou, G. Karame, and I. Christou, SuperTrust–A secure and efficient framework for handling trust in Super Peer networks, in Proc. of the 9th International Conference on Distributed Computing and Networking (ICDCN'08). Kolkata, India: Springer-Verlag, January 2008, pp. 350–362.
[18] J. R. Douceur, The Sybil attack, in Proc. of the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02). Cambridge, MA, USA: Springer Berlin / Heidelberg, March 2002, pp. 251–260.
[19] ENISA Position Paper No. 2: Reputation-based systems: a security analysis, European Network and Information Security Agency, Techical Report, October 2007. [Online]. Available: http://www.enisa.europa.eu/act/it/oar/reputation-systems/reputation-based-systems-a-security-analysis
[20] E. Fehr and U. Fischbacher, The nature of human altruism, Nature, vol. 425, no. 6960, pp. 785–791, 2003.
[21] M. Feldman, K. Lai, I. Stoica, and J. E. Chuang, Robust incentive techniques for peer-to-peer networks, in Proc. of the 5th ACM Conference on Electronic commerce. New York, NY, USA: ACM, May 2004, pp. 102–111.
[22] N. Gal-Oz, E. Gudes, and D. Hendler, A robust and knot-aware trust-based reputation model, in Proc. of the 2nd Joint iTrust and PST Conferences on Privacy, Trust Management and Security (IFIPTM'08). Trondheim, Norway: Springer Boston, June 2008, pp. 167–182.
[23] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, Reputation-based framework for high integrity sensor networks, ACM Transactions on Sensor Networks (TOSN), vol. 4, no. 3, 2008.

[24] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, Propagation of trust and distrust, in Proc. of the 13th International Conference on World Wide Web (WWW'04). New York, USA: ACM, May 2004, pp. 403–412.
[25] C.-W. Hang, Y. Wang, and M. P. Singh, An adaptive probabilistic trust model and its evaluation, in Proc. of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'08). Estoril, Portugal: International Foundation for Autonomous Agents and Multiagent Systems, May 2008, pp. 1485–1488.
[26] R. Hasan, Z. Anwar, W. Yurcik, L. Brumbaugh, and R. Campbell, A survey of peer-to-peer storage techniques for distributed file systems, in Proc. of International Conference on Information Technology: Coding and Computing (ITCC'05). Las Vegas, Nevada, USA: IEEE Computer Society, April 2005, pp. 205–213. [Online]. Available: http://dx.doi.org/10.1109/ITCC.2005.42
[27] K. Hoffman, D. Zage, and C. Nita-Rotaru, A survey of attack and defense techniques for reputation systems, ACM Computing Surveys (CSUR), vol. 42, no. 1, pp. 1–31, 2009.
[28] J. Huang and D. Nicol, A Formal-Semantics-Based Calculus of Trust, Internet Computing, IEEE, vol. 14, no. 5, pp. 38–46, 2010.
[29] A. Jøsang, An algebra for assessing trust in certification chains, in Proc. of the Network and Distributed Systems Security Symposium (NDSS'99), San Diego, California, USA, February 1999.
[30] A. Jøsang and J. Golbeck, Challenges for robust trust and reputation systems, in Proc. of the 5th International Workshop on Security and Trust Management (STM'09). Saint Malo, France: Elsevier, September 2009.
[31] A. Jøsang and J. Haller, Dirichlet reputation systems, in Proceedings of the Second International Conference on Availability, Reliability and Security (ARES 2007). Vienna, Austria: IEEE Computer Society, April 2007, pp. 112–119. [Online]. Available: http://dx.doi.org/10.1109/ARES.2007.71
[32] A. Jøsang and R. Ismail, The Beta reputation system, in Proc. of the 15th Bled Electronic Commerce Conference, vol. 160, Bled, Slovenia, June 2002.
[33] A. Jøsang, R. Ismail, and C. Boyd, A survey of trust and reputation systems for online service provision, Decision Support Systems, vol. 43, no. 2, pp. 618–644, 2007.
[34] A. Jøsang, S. Marsh, and S. Pope, Exploring different types of trust propagation, in Proc. of the 4th International Conference of Trust Management (iTrust'06), series Lecture Notes in Computer Science, Pisa, Tuscany, Italy, May 2006, pp. 179–192. [Online]. Available: http://sky.fit.qut.edu.au/ josang/papers/JPM2006-iTrust.pdf
[35] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, The Eigentrust algorithm for reputation management in p2p networks, in Proc. of the 12th International Conference on World Wide Web (WWW'03). Budapest, Hungary: ACM, May 2003, pp. 640–651.
[36] R. Kerr and R. Cohen, Modeling trust using transactional, numerical units, in Proc. of the 2006 International Conference on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services (PST'06). Markham, Ontario, Canada: ACM, October 2006.
[37] R. Kerr and R. Cohen, Smart cheaters do prosper: Defeating trust and reputation systems, in Proc. of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'09). Budapest, Hungary: International Foundation for Autonomous Agents and Multiagent Systems, May 2009, pp. 993–1000.
[38] M. Kinateder, E. Baschny, and K. Rothermel, Towards a generic trust model–Comparison of various trust update algorithms, in Proc. of the 3rd International Conference of Trust Management (iTrust'05). Paris, France: Springer-Verlag, May 2005, pp. 177–192.
[39] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, Systematic literature reviews in software engineering - A systematic literature review, Information and Software Technology, vol. 51, no. 1, pp. 7–15, 2009.
[40] N. Koblitz and A. J. Menezes, A survey of public-key cryptosystems, SIAM Review, vol. 46, no. 4, pp. 599–634, 2004.
[41] L. M. Kohnfelder, Towards a practical public-key cryptosystem. Massachussetts Institute of Technology, 1978, thesis (B.S.). [Online]. Available: http://hdl.handle.net/1721.1/15993
[42] F. Li and J. Wu, Uncertainty modeling and reduction in MANETs, IEEE Transactions on Mobile Computing, vol. 9, no. 7, pp. 1035–1048, 2010.
[43] Q. Li and K. M. Martin, A secure marketplace for online services that induces good conduct, in Short Paper Proceedings of the Fourth IFIP WG11.11 International Conference on Trust Management (IFIPTM 2010), Morioka, Iwate, Japan, jun 2010, pp. 65–72. [Online]. Available: http://www.ifip-tm2010.org/lib/exe/fetch.php?media=shortpaper09.pdf
[44] Z. Liang and W. Shi, PET: A personalized trust model with reputation and risk evaluation for P2P resource sharing, in Proc. of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05). Big Island, HI, USA: IEEE Computer Society, January 2005.
[45] J. Liu and V. Issarny, Enhanced reputation mechanism for mobile ad hoc networks, in Proc. of the 2nd International Conference on Trust Management (iTrust'04. Oxford, UK: Springer-Verlag, March 2004, pp. 48–62.
[46] L. Liu and W. Shi, Trust and Reputation Management, Internet Computing, IEEE, vol. 14, no. 5, pp. 10–13, 2010.
[47] F. G. Marmol and G. M. P´ erez, Security threats scenarios in trust and reputation models for distributed systems, Computers & Security, 2009.
[48] S. Marti and H. Garcia-Molina, Taxonomy of trust: Categorizing P2P reputation systems, Computer Networks, vol. 50, no. 4, pp. 472–484, 2006.
[49] A. Mislove, A. Post, P. Druschel, and K. P. Gummadi, Ostra: Leveraging trust to thwart unwanted communication, in Proc. of the 5th USENIX Symposium on Networked Systems Design and Implementation (NSDI'08). San Francisco, California, USA: USENIX Association, April 2008, pp. 15–30.

[50] L. Mui, M. Mohtashemi, and A. Halberstadt, A computational model of trust and reputation, in Proc. of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02). Big Island, HI, USA: IEEE Computer Society, January 2002, pp. 2431–2439.
[51] L. Mui, M. Mohtashemi, and A. Halberstadt, Notions of reputation in multi-agents systems: A review, in Proc. of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'02). Bologna, Italy: ACM, July 2002, pp. 280–287.
[52] A. Nandi, T.-W. Ngan, A. Singh, P. Druschel, and D. S. Wallach, Scrivener: Providing incentives in cooperative content distribution systems, in Proc. of the 6th ACM/IFIP/USENIX International Middleware Conference, Grenoble, France, November 2005, pp. 270–291.
[53] L. Page, S. Brin, R. Motwani, and T. Winograd, The PageRank citation ranking: Bringing order to the web, Stanford InfoLab, Techical Report, 1999.
[54] J. Patel, W. L. Teacy, N. R. Jennings, and M. Luck, A probabilistic trust model for handling inaccurate reputation sources, in Proc. of the 3rd International Conference of Trust Management (iTrust'05). Paris, France: Springer-Verlag, May 2005, pp. 193–209.
[55] R. Paul and Z. Richard, Trust among strangers in Internet transactions: Empirical analysis of eBay's reputation system, in The Economics of the Internet and E-Commerce, series Advances in Applied Microeconomics: A Research Annual, vol. 11. Elsevier, 2002, pp. 127–157.
[56] S. Reece, A. Rogers, S. Roberts, and N. R. Jennings, Rumours and reputation: evaluating multi-dimensional trust within a decentralized reputation system, in The sixth international joint conference on autonomous agents and multi-agent systems (AAMAS-07), Honolulu, Hawaii, USA, May 2007, pp. 1063–1070. [Online]. Available: http://eprints.ecs.soton.ac.uk/13260/
[57] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, Reputation systems, Communications of the ACM, vol. 43, no. 12, pp. 45–48, December 2000.
[58] M. Richardson, R. Agrawal, and P. Domingos, Trust management for the semantic web, in Proc. of the 2nd International Semantic Web Conference (ISWC'03), series Lecture Notes in Computer Science. Sanibel Island, Florida, USA: Springer Berlin / Heidelberg, October 2003, pp. 351–368. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-39718-2 23
[59] O. Roozmand, M. A. Nematbakhsh, and A. Baraani, An electronic marketplace based on reputation and learning, JTAER: Journal of Theoretical and Applied Electronic Commerce Research, vol. 2, no. 1, pp. 1–17, 2007.
[60] A. Ruiz-Martínez, D. Sánchez-Martínez, M. Martínez-Montesinos, and A. F. Gómez-Skarmeta, A survey of electronic signature solutions in mobile devices, JTAER: Journal of Theoretical and Applied Electronic Commerce Research, vol. 2, no. 3, pp. 94–109, 2007.
[61] S. Ruohomaa, A. Hankalahti, and L. Kutvonen, Detecting and reacting to changes in reputation flows, in Proceedings of IFIPTM 2011. Copenhagen, Denmark: Springer, June 2011, to appear.
[62] S. Ruohomaa and L. Kutvonen, Trust management survey, in Proc. of the 3rd International Conference on Trust Management (iTrust'05), series Lecture Notes in Computer Science, vol. 3477. Paris, France: Springer-Verlag, May 2005, pp. 77–92. [Online]. Available: http://dx.doi.org/10.1007/11429760 6
[63] S. Ruohomaa and L. Kutvonen, Trust and distrust in adaptive inter-enterprise collaboration management, JTAER: Journal of Theoretical and Applied Electronic Commerce Research, vol. 5, no. 2, pp. 118–136, 2010.
[64] S. Ruohomaa and L. Kutvonen, From subjective reputation to verifiable experiences—implementing social control in open service ecosystems, University of Helsinki, Department of Computer Science, Techical Report, March 2011, CINCO internal report.
[65] S. Ruohomaa, L. Kutvonen, and E. Koutrouli, Reputation management survey, in Proc. of the 2nd International Conference on Availability, Reliability and Security (ARES'07). Vienna, Austria: IEEE Computer Society, August 2007, pp. 103–111.
[66] J. Sabater and C. Sierra, Reputation and social network analysis in multi-agent systems, in Proc. of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'02). Bologna, Italy: ACM, July 2002, pp. 475–482.
[67] J. Sabater and C. Sierra, Review on computational trust and reputation models, Artificial Intelligence Review, vol. 24, pp. 33–60, September 2005.
[68] R. Sherwood, S. Lee, and B. Bhattacharjee, Cooperative peer groups in NICE, Computer Networks, vol. 50, pp. 523–544, March 2006.
[69] V. Shmatikov and C. Talcott, Reputation-based trust management, Journal of Computer Security, vol. 13, no. 1, pp. 167–190, 2005.
[70] A. Singh and L. Liu, TrustMe: anonymous management of trust relationships in decentralized P2P systems, in Proc. of the 3rd International Conference on Peer-to-Peer Computing (P2P'03). Linkoping, Sweden: IEEE Computer Society, September 2003.
[71] S. Song, K. Hwang, R. Zhou, and Y.-K. Kwok, Trusted P2P transactions with fuzzy reputation aggregation, IEEE Internet Computing, vol. 9, no. 6, pp. 24–34, November 2005.
[72] M. Srivatsa, L. Xiong, and L. Liu, TrustGuard: Countering vulnerabilities in reputation management for decentralized overlay networks, in Proc. of the 14th International Conference on World Wide Web (WWW'05). Chiba, Japan: ACM, May 2005, pp. 422–431.
[73] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, Chord: A scalable peer-to-peer lookup service for Internet applications, in Proc. of the 2001 Conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM'01). San Diego, California, United States: ACM, August 2001, pp. 149–160.

[74] Y. L. Sun, Z. Han, W. Yu, and K. R. Liu, A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks, in Proc. of the 25th IEEE INFOCOM, Barcelona, Catalunya, Spain, April 2006, pp. 230–236.
[75] P.-N. Tan, M. Steinbach, V. Kumar et al., Introduction to data mining. Addison-Wesley, 2006, ch. 10: Anomaly Detection.
[76] T. Tran and R. Cohen, Improving user satisfaction in agent-based electronic marketplaces by reputation modelling and adjustable product quality, in Proc. of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'04). New York, USA: ACM, July 2004, pp. 828–835.
[77] L. Viljanen, Towards an ontology of trust, in Proc. of the 2nd International Conference on Trust, Privacy and Security in Digital Business. Copenhagen, Denmark: Springer Berlin / Heidelberg, August 2005, pp. 175–184. [Online]. Available: http://dx.doi.org/10.1007/11537878 18
[78] K. Walsh and E. G. Sirer, Experience with an object reputation system for peer-to-peer filesharing, in Proc. of the 3rd USENIX Symposium on Networked Systems Design and Implementation (NSDI'06). San Jose, CA, USA: USENIX Association, May 2006, pp. 1–14.
[79] Y. Wang and M. P. Singh, Trust representation and aggregation in a distributed agent system, in Proc. of The 21st National Conference on Artificial Intelligence (AAAI'06), vol. 21, no. 2, Boston, Massachusetts, USA, July 2006, pp. 1425–1430.
[80] Y. Wang and M. P. Singh, Formal trust model for multiagent systems, in Proc. of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07), Hyderabad, India, January 2007, pp. 1551–1556.
[81] L. Xiong and L. Liu, Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities, IEEE Transactions on Knowledge and Data Engineering, vol. 16, no. 7, pp. 843–857, 2004.
[82] B. Yu and M. P. Singh, A social mechanism of reputation management in electronic communities, Cooperative Information Agents IV-The Future of Information Agents in Cyberspace, pp. 355–393, 2000.
[83] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao, Sybillimit: A near-optimal social network defense against Sybil attacks, in Proc. of the IEEE Symposium on Security and Privacy, Oakland, California, USA, May 2008, pp. 3–17.
[84] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, Sybilguard: Defending against Sybil attacks via social networks, in Proc. of the ACM SIGCOMM Conference. Pisa, Italy: ACM, September 2006, pp. 267–278.
[85] L. A. Zadeh, Fuzzy logic and approximate reasoning, Synthese, vol. 30, no. 3, pp. 407–428, 1975.
[86] R. Zhou and K. Hwang, Gossip-based reputation aggregation for unstructured peer-to-peer networks, in IEEE International on Parallel and Distributed Processing Symposium (IPDPS'07). IEEE Computer Society, March 2007, pp. 1–10.
[87] R. Zhou and K. Hwang, Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing, IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 4, pp. 460–473, 2007.