Securing P2P Networks Using Peer Reputations: Is there a silver bullet?

Prashant Dewan
Department of Computer Science and Engineering
Arizona State University
Tempe, Arizona 85281
Email: dewan@asu.edu

Partha Dasgupta
Department of Computer Science and Engineering
Arizona State University
Tempe, Arizona 85281
Email: partha@asu.edu

***Abstract*— Peer reputations have been used as security tools to not only motivate peers against *cheating* but also protect good peers from the chronic cheaters. Although the reputation management techniques are not confined to P2P networks, they present novel challenges that were absent in central server based distributed systems. We enumerate these challenges and survey the solutions proposed by the community to counter them. These challenges include, but are not limited to, peer-identification in decentralized environments, reputation metrics, storage and exchange of reputation data. Finally we survey the applications which use P2P network paradigm and therefore can benefit from the reputation systems.**

I. Introduction

Security has never been the strong point of the Internet, and with the advent of completely decentralized networks like peer-to-peer networks, sensor and ad hoc networks, the threats have only aggravated. These networks provide higher degree of autonomy to their nodes and hence are difficult to police. *Pure* peer-to-peer networks do not have any central control or central repository, while ad hoc networks may or may not have a central control. Sensor networks have a sink which is 'central' to the sensors but sensors still need to cooperate with each other to reach the sink. Hence the key to a useful network is, *cooperation among the members of a network while achieving their individual goals*. Existence of common goals among the nodes of a network or a common control managing them, alleviates the security issues, but in their absence, there is no silver bullet for securing decentralized networks.

Considerable research has been performed on *Reputation Systems* targeted at securing decentralized systems. Many researchers have proposed [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11] the use of reputation techniques in order to motivate the peers in a peer-to-peer network to cooperate, to motivate them to abstain from cheating, and penalize the peers who cheat. This body of research work has shown that the use of reputation systems in P2P networks considerably reduces the number of malicious transactions, in decentralized networks, under certain assumptions. Reputations have been examined [12], [13], [14] from the perspective of ad hoc networks in order to motivate the network nodes to participate in routing and stay away from explicitly routing incorrectly. In ad hoc networks, reputation techniques increase the goodput of the network (the percentage of the packet sent that actually reach the target without getting maliciously altered), while the fear of banishment motivates the nodes against cheating. Although we did not find any research explicitly focused on use of reputation in the sensor networks, most of the reputation schemes designed for ad hoc networks can be chiseled to fit into the sensor network arena.

In this paper we present the components and challenges in any reputation system while using a P2P network as the underlying network. In Section II we present the main components of any reputation systems. Section III enumerates the challenges and approaches of designing a reputation system while Section IV presents the applications which can benefit from reputation systems.

II. Nuts and Bolts of Reputation Systems

We decompose the reputation of an entity into recommendations for simplicity of exposition. A *recommendation* is the reputation an entity earns for one transaction. The reputation of an entity, is the average of all the recommendations received by the entity. In all peer-to-peer reputation systems, irrespective of the fact whether they are centralized or decentralized, the number of members in the network (as long as there are at least two members), the configuration, topology, function, or the use of the network,the following components are always present(unless explicitly stated otherwise).

A. Provider or Owner:

Provider is the entity that is being evaluated for a service provided by it or an attribute possessed by it. Reputation is always associated with an entity (e.g. a peer or a file or a process). The provider *owns* the reputation. The reputation might be composed of one recommendation from one peer, or multiple recommendations from multiple peers, or any other combination of the two. In a P2P network any peer can take the role of a provider.

B. Context or Service or Attribute:

The reputation of a peer is based on a certain service it provides, or pertains to a certain context or an attribute it possesses. For example, a peer's reputation might be a function of the reliability of its web server (attribute), or a goodness of

Fig. 1. Components of a Reputation System

the content hosted on the server (context), or just the efficiency of the peer itself (service). The reputation is unlikely to be one single value, it is likely to be comprised of a set of values associated with a set of contexts. In this paper we only consider one context for each peer in order to keep things simple.

C. Requester or Evaluator:

A requester requests the service from the provider and optionally evaluates the attributes of the provider besides giving recommendation(s) to the provider. The method used by the requester to ascertain the value of the recommendation it provides, can be common across peers (e.g. all peers give a recommendation 1.0 if they are able to download a file irrespective of the content file) or specific to a peer (e.g. every peer evaluates the file and then decides on the recommendation). In a completely decentralized network, the method used to ascertain the recommendation is unlikely to be global, because unrelated peers are likely to value the different aspects of a service differently.

A requester also has the choice to examine the reputation of a given provider before it decides to perform a transaction with the provider. Again, the method it uses to calculate the reputation from the set of recommendations of the provider can either be global or modeled as per the goals of the individual peer.

D. Transaction:

When a requester receives a service from a provider it completes a transaction. A transaction can be as simple as a file-download or as complicated as a multi-year contract of a set of services. Transaction should be atomic such that at the end of the transaction, either the provider has provided the service and received the recommendation or neither of the two has been done. This is important because a half-cooked transaction can be used to question the credibility of the provider as the two parties in the transaction (requester and provider) would report the transaction differently to a future requester.

E. Quantitative or Subjective:

A recommendation can be quantitative (e.g. 4 out of 10) or subjective (e.g. Fine, OK, Good, Very Good) or subjective with a quantitative interpretation (e.g. Good=5, V.Good=7.5, Excellent=10). Due to the nature of computers it is almost impossible to accumulate subjective metrics. If a peer has received recommendations of {Good, Bad, V. Good, V. Good, V. Bad} what should be the final conclusion? This question cannot be answered without quantitative interpretation of the assigned subjective recommendations (unless of course all the recommendations of a peer are V. Good).

F. Metrics and Accumulators:

One of the most important components of the quantitative reputation systems is the metrics used for the reputation. Websites like eBay use (-1,0,+1) system of exchanging recommendations. The recommendations are subsequently accumulated using statistical accumulators like MEAN, MEDIAN etc. in order to calculate the reputation of the provider. Probability techniques can also be used for quantification of recommendations. Probabilistic techniques are a more accurate way of representing uncertainty than statistical techniques. A reputation of p possessed by a provider would not mean much to a future requester but a probability (of fair transaction), p would surely be useful. Some reputation systems use the Bayesian form of probability [15] while others use the Dempster-Shafer form [16]. The pros and cons of various reputation metrics have been elaborated in Section III-B

G. Storage:

The reputation information has to be stored in the network. Storage is an important component but not a mandatory part of a reputation system. Theoretically all peers might decide to *remember* the outcome of their transactions and present them on request. Hence the reputation information would not have to be stored anywhere in a small and ideal P2P network. Social networks do not store reputation information anywhere. In the scenario where the availability of the peers is erratic, peers might lie, or they might simply forget the reputation information of the past transactions, the information not only has to be stored at a *secure* location, but also stored in a fashion such that its untamperable and non-repudiable by the related peers. The locations where the reputation can be possibly stored is with the provider, with the requester or a third party in a network.

III. Challenges and Approaches

Reputation Systems for completely decentralized networks pose a unique set of challenges. Some of the issues like identity management and storage management are not new and have been extensively studied in the realm of distributed systems. These issues become more challenging when confronted as a part of design and implementation of a security infrastructure, as they are the cornerstones of security solutions like reputation systems.

A. Identity Management

The reputation of a peer is generally associated with the *identity* of the peer. Identity can be a digital certificate or a pseudonym that can uniquely identify a peer. The fundamental questions that need to be answered for identity management in P2P reputation systems are: Should a peer have one or multiple identities? Who allocates the identities to the peers? What all information should an identity of a peer contain? Should one identity lead to only one peer or it can identify two (or more) unrelated peers? In the following paragraph we attempt to answer some of these questions.

Identity of a peer in a P2P network has to be unique to a peer or a set of peers. If two peers claim to be Alice and do transactions separately this would disrupt the reputation system because it would provide an incorrect view of the reputation of Alice to a prospective requester. The remedy to this issue is either the two peers should not be allotted the same identity or their recommendations should be merged to calculate the reputation of the identity, Alice. The two different peers behind the identifier, Alice, will become transparent to the requester.

A peer in a P2P network can have one or more identities. The concept of multiple identities resonates with the identity management in real life where human beings have multiple identities (SSN, License no., Passport no.). In addition, multiple identities enable a peer to be anonymous in one transaction, *pseudononymous* in another and identify itself in the third transaction which is extremely useful in P2P networks. On the negative side, multiple identities can be easily misused. A peer with multiple identities can use a subset of its identities to provide recommendations to its other identities and thereby raise their reputation. Hence a peer holding multiple unrelated identities can effectively raise its own reputation without even doing a single transaction with any other peer. This foils the security and the usefulness of a reputation system.

The set of identities owned by a peer used to raise the reputation of another set of identities owned by the same peer is called a *liar farm* [17]. There are no good solutions to counter liar farms in a decentralized network. Introduction of a central component like a certificate authority or an authentication server can mitigate the threat of a liar farm (at the cost of flexibility) by making sure that either one peer does not get more than one identity, or the identity set of a peer is made such that two identities from the same set can be linked to each other. This takes away the anonymity from a peer although it can still be pseudononymous- each of its identities can be linked to each other but not to the peer. Allocation of multiple sets of identities to a peer does not provide anonymity, it only makes the problem recursive because the authority that allocates the identity can still map the identity to the peer. A brief overview of various identity management issues in decentralized networks is given in [18].

As we said before, in decentralized networks there is no Certificate Authority (CA). Therefore the peers have to generate their own identities. Only *resource constraints* can stop a peer from generating multiple identities. Given the technological advances in computing, bandwidth and storage, resource constraints are only weak fetters for malicious nodes. Resource constraint techniques based on micropayments not only waste resources, they are only limited by the technical knowhow and the buying power of the peer.

Self Certification [17] not only obviates the need of a trusted central server but also provides greater autonomy to the peers in the network. Self-certification facilitates the peer to generate their own identities. The peers run their own CA and generate identities for themselves. As a result a given peer generates as many identities as it wants to. This raises the obvious problem of *liar farms*. The threat of liar farms is mitigated with *IP Based Safeguard*(IBS) [17], which is based on the fundamental premise that a given peer cannot have access to a large number of discontinuous IP spaces. In addition, the IBS assumes that the peers in the network are more interested in the relative ranking of the other peers and less interested in their absolute reputations. IP Based Safeguard is inspired from the techniques Google uses to protect its ranking mechanisms from users who try to jack up the ranks of their pages by putting to and fro links in them.

Hauswirth *et al.* have proposed a P2P based directory system in [19]. In their directory system the peers generate their own identifiers and insert the identifier-location tuple in the decentralized network. Any other peer searching for the a particular peer, searches the network with the identifier of the peer and obtains the tuple containing the location. This system works well for most of the identity management needs but does suffers from the problem of liar farms (if used in a reputation system).

B. Metrics

A reputation is a highly subjective and epistemological attribute of an entity. The current literature has predominantly dealt with the issue of combining recommendations to generate reputations with either a statistical viewpoint, using regression analysis, median, mean or a probabilistic viewpoint, using the Bayesian theory. Although the *empirical rule* on the data following the bell-shaped curve does encapsulate the range of uncertainty [20] statistically, it uses statistics to generate probability values. Therefore probability appears to be a better choice for representing uncertainty as it can represent uncertainty more accurately than the statistical techniques. A large fraction of current generation of reputation systems are text or comment based reputation systems where the reputation issuers give comments, along with (or instead of) numerical

ratings. We do not discuss the techniques for processing these comments here. On line reputation systems such as eBay and Amazon use statistical techniques to capture information of providers' past transactions as rated by the past requesters. In eBay, the providers receive feedback (-0,0,+1) from the requesters. The feedback received by a provider is arithmetically accumulated to gauge the trustworthiness of a provider.

Interestingly, the *Bayesian Theory of Probability* does not lend itself well to reputation systems. The objectivists, following the Bayesian theory, try to find out the repeated frequency of a sequence of events in a set of events. They use prior experiments in highly controlled conditions, to find such repeated patterns. The subjectivists on the other hand try to encompass subjectivity with objective interpretation and expect the subject to not only be aware of the factors influencing a transaction but also to be able to put a bet on it. In a reputation system, the requester can neither perform such experiments, nor can it enumerate the factors effecting the transaction, let alone putting bets on them. Alice, a requester in the transaction, cannot predict the result of the transaction unless it understands all the motivating factors that are influencing the provider, Bob. These factors would finally influence the transaction. Hence, neither of the above two theories fits the requirements of reputation systems.

Besides the above-mentioned limitations, Bayesian models do not allow the peers to represent the state where a peer 'does not know' about the other peer. On the other hand, the 'does not know' state can be easily represented using the belief functions as reported by Dempster and Shafer [21]. Depending on the metrics used sophisticated probability accumulators are needed to add probability based recommendations. All peers should use the same accumulator in order to achieve consistent results, although peers are free to use different accumulators where consistency of reputation information is not a goal.

Dempster-Schafer calculus appears to represent reputation most accurately. Dempster-Shafer calculus can be used for calculating reputations from recommendations. The use of Dempster-Shafer calculus for reputation management has been proposed by Dewan *et al.* [17]. In addition Yu *et al.* have experimentally shown the benefits of using the Dempster-Shafer calculus in reputation systems. In [17] *Belief Functions* are used to assign numeric values to recommendations and the *Dempster's rule of Combination of Belief functions* is used an accumulator to calculate the reputation. In the reputation system based on Dempster-Shafer calculus, a recommendation received by a peer is represented by *degrees of belief*. A peer has one belief for a question and it divides that belief into a subset of propositions. For example, a peer can divide his belief on the outcome of an eBay transaction in which it wants to buy a computer into three propositions:

1) The seller **will send** the computer after taking the money, Bel(fair) = 0.5
2) The seller **will not send** the computer after taking the money, Bel (cheat1)= 0.3
3) The computer **will not work** Bel (cheat2) =0

The above statements imply that the peer has the evidence to estimate the likelihood of the fact that the seller will send the computer. The peer does not have any evidence to ascertain whether the computer will work or not. This does not mean that he believes that it is equally likely that the computer may or may not work. It only means that the seller has no information to decide whether the computer will work. The above example can be further elaborated as:

Let the question, *Q = Will the seller cheat in the transaction*?

Suppose the requester has an evidence from two other peers, Alice and Charles. Alice and Charles both say that the seller had not cheated when they transacted with him. Now the requester needs to consider if one or both of Alice and Charles are lying. Let

D = Alice is truthful D' = Alice is lying D = 0.8
{D,D'} =Unsure
E = Charles is truthful E' = Charles is lying E = 0.9
{E,E'} =Unsure

D⇓ E ⇒		E	E'	{E,E'}
		0.9	0	0.1
D	0.8	0.72	0	0.8
D'	0	0	0	0
{D, D'}	0.2	0.18	0	0.02

- Bel (Q) = The seller will not cheat in the transaction = 0.72 + 0.08 + 0.18 = 0.98
- Bel (Q') = The seller will cheat in the transaction = 0
- Bel (Unknown) = 0.2

The following properties of the reputation metrics can be observed from the above example:

1) The calculation of the reputation in the above form allows the user to express indecisiveness.
2) It can be calculated incrementally.
3) It is commutative.

C. Storage

One of the important components of any reputation system is the storage of reputation data. It assumes such high importance because the security of the system is dependent on the integrity of the data, the format it is stored in, and the location of the storage. In a decentralized P2P network, there are only three choices for the location of reputation data: 1) the requester 2) the provider or 3) it might be stored with some third party in the network. The third party might be selected at random or by mutual agreement between the provider and the requester.

Before we evaluate the storage choices lets look at the kind of reputation data available in P2P networks. The body of the reputation literature can be categorized into two main groups: 1) reputation systems in which peers use only their own experience *(local information)* for evaluating other peers [17], 2) systems in which peers use the experiences of other peers

(global information) [22]. In the absence of any central authority the global information is generated from the local information (managed by each peer), using various decentralized schemes [10]. The global information is highly susceptible to peers that falsify their local information. Although the local information is more trustworthy for a peer, the global information considerably speeds up the process of identification of malicious peers, as peers learn from each other's transactions.

Therefore the reputation data, both global and local information, have to be protected from the malicious third parties besides protecting it from the requester and the provider. A logical solution to this requirement is to store the reputation data with a mutually agreed third party which has no interest in the data and hence will not possess any motivation to maliciously tamper with the data. The drawbacks of this approach are that (at least theoretically) a third party can be compromised, the third party might not be available (due to the erratic availability of peers in a P2P network) when needed or may not even care to store the data securely. The solution might lie in storing the data with multiple third parties and using cryptographic techniques to protect the data. Finding a large number of mutually agreeable third parties is a difficult process, hence the set of third parties will have to be selected randomly. This mitigates the possibility of the compromise of the third party but when coupled with the erratic availability of the peers, raises the threat because now the attacker has a choice to attack the weakest subset out of the set (comprising the third party). Above all, the inclusion of the third party breaches the fundamental peer-to-peer paradigm.

Trust chains are also popular in establishment and propagation of trust among peers. The chain of trust are based on the basic premise that the trust is transitive. Many trust based systems like Advogato [23], PGP [24] & Chain of Trust [25] use the concept of chains. While Advogato does not explicitly talk about storage, PGP certificates may be stored with the peers or at a central server. Hence the remaining choices are storing the data with either the requester or the provider. The data can be stored with the requester and protected cryptographically but the problem is that future requesters for a given provider will have to contact all the past requesters (of the given provider) in order to fetch and validate the recommendations before interacting with a given provider. A P2P network is already bandwidth intensive and search in an unstructured P2P network is extremely expensive.

Hence the appropriate choice of location for reputation data is with the provider itself. The future requesters will find the reputation data in one place and will not have to search the network for the data. The provider would protect the data because it has a stake in the data. Hence the only challenge left is to protect the data from the provider itself. A two party cryptographic protocol can facilitate the data storage with the provider itself while protecting it from the provider using cryptographic techniques. In the protocol proposed by Dewan *et al.* in [26], cryptographic blinding techniques coupled with digital signature and symmetric key encryption are used to create a chain of recommendations of a given provider. The sequence number of the last recommendation in the chain is public knowledge. As a result, the provider cannot append any entry to the chain and it cannot modify any entry in the chain because all entries are signed by the past requesters. In addition, the provider is forced to commit to a recommendation value before it can see the actual value using cryptographic blinding techniques. The requester does not have to search for the past recommendations of the provider as they are available with the provider itself. Storing the reputation data with the provider takes away the need for traversing chains of recommendation in order to establish the credibility of the sources.

As mentioned in Section II-G, storage is not a mandatory component of a reputation system. In the P2PRep network, reported by Damiani *et al.* [9] a peer looking for the reputation of another peer polls the neighbors of the possible provider and takes positive or negative votes from them. Hence the requester does not have to store the reputation information of the provider. In RCert [27] reported by Ooi *et al.*, the reputation data is stored with the provider but a third party is used to ensure the integrity and the authenticity of the data.

D. False Recommendations

In any bilateral protocol the party who moves second has a temptation to cheat i.e. to defect away from the agreed terms of the transaction in such a way that it benefits at the cost of the other party [8]. Hence bilateral protocol that involves the exchange of reputation has to be either supervised by a third party or it has to be designed in such a way that the neither of the two parties gets an advantage by the virtue of the order in which they perform their roles. For example, the requester should not be able to give a bad recommendation when it has received an excellent service, *Bad Mouthing* The requester should not give a good recommendation to a bad peer only because the bad peer is requester's friend, *Ballot Stuffing*. Similarly the provider should not discriminate against requesters.

This problem is extremely difficult to solve because it is very difficult to stop a peer from lying. Presence of a third party that evaluates the relationship between the quality of the service provided by the provider and the corresponding recommendation received by the provider can possibly solve this problem. This would only work under the assumption that the third party is trustworthy and uncompromisable and available at the time of the transaction. Dellarocas *et al.* have proposed anonymity of the provider and the requester in order to solve the problems of discrimination, ballot stuffing and bad mouthing in [28]. Although pseudononymity can mitigate the threat but anonymity does not work in reputation systems because identity is the first requirement of a reputation. The other reputation systems like RCert, P2PRep etc., the cryptographic protocol proposed by Dewan *et al.* ignore this problem.

IV. Applications

Reputation systems have a myriad set of applications. Users of most of the applications rank other users on the basis of

their reputations and select one or more user or entity to perform the transaction. Once the transaction is complete the requester gives a recommendation to the provider and the recommendation becomes the part of the reputation of the provider.

A. Market Place

There are plenty of websites which use *collaborative filtering*, to recommend products to their buyers. Collaborative filtering works on the simple premise that if Alice likes music and cooking and Bob like music then most likely Bob would also like cooking. This logic is used by the websites like Amazon to recommend products to its customers. GroupLens [29], a research project in the University of Minnesota uses automated collaborative filtering to recommend movies to its patrons. The other form of reputations is used in marketplaces like eBay. In marketplaces like eBay and Amazon, the users buy and sell goods on the basis of the reputation of the other party. The fundamental premise in all these websites is that the company managing the website like Amazon and eBay would not cheat because they have a reputation to live by.

B. Trust Management

Reputations have been predominantly used for trust management. Here we define the word *trust* as the probability (in a loose sense) that the trusted party would meet the expectations of the trusting party. Reputations cannot be used to generate absolute trust, they can only be used to minimize the risk. Lets assume that Alice has never cheated in the past 1000 transactions that she has performed. In addition, all the other peers who have performed transactions with Alice have been extremely satisfied. Therefore it is only *likely* that Alice will not cheat in the next transaction. There is no absolute surety that she would not. The likelihood is increased by the fact that Alice would not to see her reputation reduced. This premise only works if Alice still values her reputation as she has done in the past. If she stops valuing her reputation then she is likely to cheat for other benefits. A large number of researchers have shown that (at least in experiment scenarios) use of reputation motivates peers to abstain from cheating and reduces the number of malicious transactions from the perspective of the whole system and the perspective of each peer by more than half.

C. Increased Throughput

Reputations have been extensively used in ad hoc networks in order to counter nodes which do not fulfill their promise to other nodes [14], [12], [13], [30]. Nodes in mobile ad hoc networks have a limited transmission range. Hence the nodes expect their neighbors to relay packets meant for far off destinations. These networks are based on the fundamental assumption that if a node promises to relay a packet, it will relay it and will not cheat. This assumption becomes invalid when the nodes in the network have tangential or contradictory goals.

The reputations of the nodes, based on their past history of relaying packets, can be used by their neighbors to ensure that the packet will be relayed by the node. Instead of choosing the shortest path to the destination, the source node chooses a path whose next hop node has the highest reputation. Dewan et al. in [30] have reported that the recursive use of reputations, in the presence of 40% malicious nodes, improves the throughput of the system to 65%, from 22% throughput provided by AODV. This improvement is obtained at the cost of a higher number of route discoveries with a minimal increase in the average hop length. The other research on the use of reputation in ad hoc networks, OCEAN [14], CONFIDANT [13], CORE [12], Nuggets [31], and SPRITE [32] which are built on *Dynamic Source Routing* [DSR] and make the nodes to snoop into the activities of their neighbors. While SPRITE is based on a central server approach, OCEAN works well as long as there are no collusions among nodes. All these systems reduce the number of malicious transactions.

V. Conclusions

Reputation systems can be used for securing decentralized networks. The absence of central trusted authority poses formidable challenges which need to be handled before reputation systems would be effective in securing decentralized networks. The solution lies in using the right recipe of components appropriate for a given application. We reiterate, that reputation systems are not a panacea for all trust problems or security issues in decentralized networks, they can only be used to mitigate the threats faced by such networks. In addition reputation systems can be used to motivate the peers to contribute to the system and weed out the malicious peers.

References

[1] Y. Wang, "Bayesian network-based trust model in peer-to-peer networks," in *Workshop on "Deception, Fraud and Trust in Agent Societies" at the Autonomous Agents and Multi Agent Systems 2003 Conference (AAMAS-03)*, Melbourne, Australia, July 2003.

[2] M. Venkatraman, B. Yu, and M. P. Singh, "Trust and reputation management in a small-world network," in *Fourth International Conference on MultiAgent Systems*, Boston, MA., july 2000, pp. 449–450.

[3] S. Marti and H. Garcia-Molina., "Identity-crisis: Anonymity vs. reputation in P2P systems." in *Third IEEE International Conference on Peer-to-Peer Computing*, 2003.

[4] H. Lee and K. Kim, "An adaptive authentication protocol based on reputation for peer-to-peer system," in *Symposium on Cryptography and Information Security*, Itaya, Japan, 2003, pp. 661–666.

[5] B. Gary, E. K. Elena, and O. Axel, "How effective are online reputation mechanisms? an experimental investigation." *Discussion Papers on Strategic Interaction*, 2002.

[6] R. Dingledine, N. Mathewson, and P. Syverson, "Reputation in P2P anonymity systems," in *Workshop on economics of P2P systems*, June 2003.

[7] C. Dellarocas and P. Resnick, "Online reputation mechanisms: A roadmap for future research," MIT, Tech. Rep., 2003.

[8] C. Dellarocas, *Building trust on-line : the design of reliable reuptation mechanism for online trading communities.* Cambridge, Ma.: MIT Sloan School of Management, 2001.

[9] E. Damiani, D. C. di Vimercati DEA, S. P. DEI, P. S. DTI, and F. V. DEI, "A reputation-based approach for choosing reliable resources in peer-to-peer networks," in *Conference on Computer and Communications Security (CCS 02)*. Washington, DC, USA: ACM Press, 2002, pp. 207–216.

[10] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in P2P networks," in *Twelfth International World Wide Web Conference*, New York, 2003, pp. 640–651.

[11] M. Chen and J. P. Singh, "Computing and using reputations for internet ratings," in *Proceedings of the 3rd ACM conference on Electronic Commerce*. ACM Press, 2001, pp. 154–162.

[12] P. Michiardi and R. Molva, "CORE: A COllaborative REputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Communication and Multimedia Security*. Portoroz, Slovenia: IEEE, September 2002.

[13] S. Buchegger and J.-Y. L. Boudec, "Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes — Fairness In Dynamic Ad-hoc NeTworks," in *Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, Lausanne, CH, June 2002, pp. 226–236.

[14] M. B. Sorav Bansal, "Observation-based cooperation enforcement in ad hoc networks," Stanford University, Tech. Rep., 2003.

[15] J.-Y. L. B. Sonja Buchegger, "A robust reputation system for P2P and mobile ad-hoc networks," in *Second Workshop on Economics of Peer-to-Peer Systems*, June 2004. [Online]. Available: http://www.eecs.harvard.edu/P2Pecon/program.html

[16] T. Yu, M. Winslett, and K. E. Seamons, "Automated trust negotiation over the internet," in *6th World Multiconference on Systemics, Cybernetics and Informatics*, Orlando, Fl., 2002.

[17] P. D. Prashant Dewan, "Countering identity farms in reputation systems for P2P networks," Arizona State University, Tech. Rep., 2004.

[18] P. Dewan and P. Dasgupta, "Identity and privacy in the virtual world," Arizona State University, Tech. Rep., 2003.

[19] M. Hauswirth, A. Datta, and K. Aberer, "Handling identity in peer-to-peer systems," in *6th International Workshop on Mobility in Databases and Distributed Systems, in conjunction with the 14th International Conference on Database and Expert Systems Applications*, September 2003.

[20] D. Rumsey, *Statistics for Dummies*. J. Wiley and Sons, 2003.

[21] G. Shafer and J. Pearl, *Readings in uncertain reasoning*. San Mateo, Calif.: Morgan Kaufmann, 1990.

[22] E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati, "Managing and sharing servents' reputations in P2P systems," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 15, pp. 840–854, July 2003.

[23] R. Levien, "Advogato," Webpage, 2003. [Online]. Available: www.Advogato.org

[24] P. Zimmermann, *The official PGP user's guide*. Cambridge, Ma.: MIT Press, 1995.

[25] B. Schneier, *Applied cryptography : protocols, algorithms, and source code in C*. New York: Wiley, 1996.

[26] P. D. Prashant Dewan, "Securing reputation data in peer-to-peer networks," in *International Conference on Parallel and Distributed Computing and Systems (PDCS 2004)*. IASTED, November 2004.

[27] B. C. Ooi, C. Y. Kiau, and K.-L. Tan, "Managing trust in peer-to-peer systems using reputation-based techniques," in *The 4th International Conference on Web Age Information Management*. LNCS, August 2003, http://xena1.ddns.comp.nus.edu.sg/P2P/waim03.pdf.

[28] C. Dellarocas, "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior," in *ACM Conference on Electronic Commerce*, Minneapolis, MN, October 2000, pp. 150–157.

[29] P. Resnick, N. Iacovou, M. Suchak, P. Bergstorm, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *ACM 1994 Conference on Computer Supported Cooperative Work*, Chapel Hill, North Carolina, August 1994, pp. 175–186.

[30] P. Dewan, P. Dasgupta, and A. Bhattacharya, "On using reputations in ad hoc networks to counter malicious nodes," in *QoS and Dynamic Systems, (in conjunction with IEEE ICPADS)*. IEEE, July 2004.

[31] L. Buttyán and J. P. Hubaux, "Enforcing service availability in mobile ad-hoc wans," in *1st ACM international symposium on Mobile ad hoc networking and computing*. Boston, Massachusetts: ACM Press, 2000, pp. 87–96.

[32] S. Zhong, J. Chen, and R. Yang, "SPRITE: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *IEEE INFOCOM*. San Francisco, USA: IEEE Press, 2002.