Open access • Journal Article • DOI:10.1504/IJCNDS.2012.047897

Analytical evaluation of P2P reputation systems — Source link

Brent Lagesse

Institutions: BBN Technologies

Published on: 01 Jul 2012 - International Journal of Communication Networks and Distributed Systems (Inderscience Publishers)

Topics: Reputation

Analytical Evaluation of P2P Reputation Systems

Brent Lagesse*

Cyberspace Science and Information Intelligence Research Group
Computational Science and Engineering Division
Oak Ridge National Laboratory
E-mail: lagessebj@ornl.gov
*Corresponding author

Abstract:
Despite widespread use of reputation mechanisms in P2P systems, little has been done in the area of analytical evaluation of these mechanisms. Current approaches for evaluation involve simulation and experimentation. These approaches provide evaluation of the mechanism in a few settings in which the experiment is designed; however, it is difficult to use these simulations for direct comparison of reputation mechanisms over a large number of systems and attacker models. In this paper, we present several analytical metrics and a utility-based method for evaluating reputation mechanisms. Further, we provide a case study of an evaluation of the EigenTrust reputation mechanism to demonstrate the use of these metrics and methods.

Keywords: P2P Systems, Security, Evaluation

Reference to this paper should be made as follows: Lagesse, B. (xxxx) 'Analytical Evaluation of P2P Reputation Systems', *Int. J. Communications Networks and Distributed Systems*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Brent is a Cyber Security Research Scientist in the Cyberspace Science and Information Intelligence Research (CSIIR) group at the Oak Ridge National Laboratory. He graduated from The University of Texas at Arlington with a PhD in Computer Science in 2009. His research focuses include security in dynamic, resource-constrained, and heterogeneous distributed systems.

1 Introduction

Reputation mechanisms are used for a wide variety of applications to predict with which entities the peer is likely to interact best. In peer-to-peer (P2P) systems, they have been used extensively as a form of security for access to resources. Reputation mechanisms are typically designed to prevent malicious or unreliable peers from providing invalid or malicious resources. Further, some reputation mechanisms are

used to prevent freeloading by giving preference to peers who are known to provide valid and useful resources.

The difficulty with implementing effective reputation mechanisms is that the mechanism itself introduces vulnerabilities into the system. Reputation mechanisms provide opportunities for attackers to game the system and manipulate the results that are presented to the user. For example, if an attacker can lie about reputation, it can promote itself as highly reputable peer and benign peers as poorly reputable peers. Additionally, an attacker can disrupt the communication that is required to transmit reputation information and prevent negative information about itself from arriving at a peer it wants to attack.

Reputation mechanisms are traditionally tested through simulation or experimentation; however, little work has been done in establishing general analytical metrics for reputation systems. Mundinger and Le Boudec [2008] analyzes liars in reputation mechanisms for mobile ad-hoc networks, but their work does not generalize to all reputation mechanisms. Dellarocas [2006] point out efficiency bounds and design principles of distributed reputation mechanisms as an important open area of research. This paper introduces metrics for reputation mechanisms (not to be confused with metrics for reputation, for example Srivatsa et al. [2005]) and an extensible framework for mathematical analysis of reputation mechanisms. Metrics for reputation mechanisms do not focus on how reputation is described in the mechanism, but rather they focus on the effectiveness of the reputation mechanism in selecting trustworthy resources. This paper describes two metrics, *accuracy* and *convergence* along with a utility model, and demonstrates their effectiveness in conjunction with our evaluation framework through a case study analysis of the EigenTrust reputation mechanism (Kamvar et al. [2003]) in Section 5.1.

Naturally, metrics do not encompass the whole of a reputation mechanism. There are many qualitative properties of reputation mechanisms such as those discussed in Hoffman et al. [2009]. The purpose of this paper is not to reduce the decision of which reputation mechanism to use to a mathematical equation, but rather to provide metrics to quantify aspects of reputation mechanisms and assist in the selection and development of reputation mechanisms for use in different types of systems and against different types of attackers.

2 Background

P2P systems involve peers that act as both clients and servers of resources. In this paper, we use *resource* as a generic term that could mean anything provided by a peer, such as a file, a printer, or a software service. These systems rely on each peer contributing resources in order for the overall utility of the system to increase. When a peer provides a faulty resource, whether unknowingly or maliciously, the utility provided by a P2P system decreases. In the context of a P2P system, it is the goal of a reputation system to increase the overall utility of the P2P system. In this sense, resource access security is achieved when the reputation system enables the P2P system to provide the same quality of service as the P2P system without any malicious peers. It is not theoretically possible to achieve this in a non-trivial utility and attack models since any reputation system will add overhead in

terms of processing or communication, but under certain attack and utility models, significant improvements can often be achieved.

Dozens of reputation systems have been proposed in research literature and in practice. Some of these systems include centralized storage of reputation information (Ebay [2009]) others acquire reputation information on demand from the originating peer (Walsh and Sirer [2006]) while others store all reputation about a peer at a peer that is unlikely to be the originating peer Kamvar et al. [2003]. Many reputation mechanisms compute local reputation values (Nandi et al. [2005] and Walsh and Sirer [2006]) and others compute a global trust value (Ebay [2009] and Kamvar et al. [2003]). Despite this diversity, there is an underlying model that all reputation mechanisms follow (Aberer and Despotovic [2001], Lagesse et al. [2009] and Hoffman et al. [2009]). In some way, the information must be acquired (for example, through experience or through communication with other peers) and a reputation must be calculated (for example, through summing all of the positive votes for a peer). Reputation mechanisms include other aspects, such as the presentation of the information and the method of making decisions based on reputation, but acquisition and calculation appear across all reputation mechanisms. These common components will be the basis of our analysis of reputation mechanisms.

As mentioned in Section 1, there are many well-known attacks against reputation mechanisms. Common attacks include sybil attacks, collusion attacks, whitewashing attacks, and denial of service attacks. Sybil attacks result from an attacker creating a large number of peers and injecting them into the system. These attacks can be used to both influence reputation calculation and affect the acquisition of reputation information. Collusion attacks involve malicious peers working together to raise their own reputation values or to lower the reputation values of other peers. For example, if a reputation mechanism weights the selection of a peer based on their reputation, then two peers could collude so that one peer acquires a high reputation value by being honest and then recommends the second peer, which acts maliciously. Whitewashing attacks occur when a peer leaves the system, and then returns (often under a new username) to attempt to lose any negative reputation that it had acquired. Denial of service attacks can be used by malicious peers to prevent a benign peer from transmitting, receiving, or computing reputation information. For example, a malicious peer could drop any reputation information passed through it that hurts its reputation. Each reputation mechanisms has a set of attacks that it performs well against and a set of attacks to which it is vulnerable.

3 Models

This paper proposes two models utilized in evaluation. The evaluative models are used to produce a specific metric. One model describes the peers in the system and another describes the reputation data in the system. The models are sufficiently generic so that they can be extended to evaluate complex behaviors and additional aspects of security.

The evaluative model is designed to determine average-case performance of the P2P system.

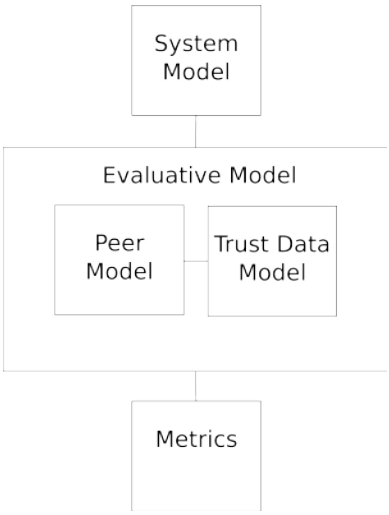

Figure 1 Evaluation Framework

Table 1 Table of Example Model Terms

P_{att}	Probability of attack
f_{ben}	Benign failure rate
f_{mal}	Malcious failure rate
M_{dep}	Departure rate of mobile peers
M_{arr}	Arrival rate of mobile peers
C	Connectivity of peers

3.1 Peer Model

The peer model describes the behavior of peers in the system. At the highest level, the peer model describes the population of peer classes in a system. The peer model describes how different classes of peers compose the system. The peer model may be dynamic and change over time. It is described by a class distribution function over the execution time of the system. The peer model can consist of several peer classes. These classes define the individual peer behavior types. There may be many different classes since a system may have many classes of attackers or benign peers or some even hybrid peers. The peer behavior class defines the behavior of a node when responding to a request for reputation information. In this section we describe two basic types of peers, benign and malicious.

The peer model classes consist of a mathematical description of the behavior of the peer in terms of its response to requests for reputation information and its computation of reputation. To see how these classes are described in practice, see the case study in Section 5.

3.1.1 Benign Class

The benign class describes the behavior of benign peers in the system. Generally these peers will correctly describe their interactions and correctly answer queries about other peers; however, this is not always the case and some benign peers may be faulty in that they sometimes fail with no malicious intent. As a result, benign classes will tend to exhibit mostly correct behavior with a small error value. For example, a simple benign class could be described by Equation 1.

$$P_{att} = f_{ben} \tag{1}$$

3.1.2 Malicious Class

The attacker model class describes the behavior of malicious peers in the system. These attackers may be colluding or acting individually. Attackers are described by their behavior in terms of their attack probability and (when relevant) how they report reputation information about other peers at a given system state. For example, a simple benign class could be described by Equation 2.

$$P_{att} = f_{mal} \tag{2}$$

3.2 Reputation Data Model

The reputation data model describes whether or not the data is accessible within the necessary time frame. This means the model can be extended to include many factors of importance such as delay tolerance, mobility, and intermittent connectivity. The reputation data model is used to determine the reputation convergence. In its most basic form, the reputation data model is used to determine the residual reputation value in a reputation computation. For example, the reputation data model could include information such as Equations 3 - 5 where the departure rate, arrival rate, and average connectivity of peers in the system are constants, independent of time.

$$M_{dep}(t) = m \tag{3}$$

$$M_{arr}(t) = a \tag{4}$$

$$C(t) = c \tag{5}$$

4 Metrics

Components of reputation mechanisms should be designed to either increase the accuracy of the mechanism or increase the speed at which the values converge. As a result, the two main domains of metrics introduced in this section are *reputation accuracy* and *reputation convergence*. This section also introduces *reputation effectiveness*, a composite metric that is based on information from the two main metric domains. Furthermore, we use utility modeling to further extract information about reputation mechanisms. Utility modeling allows us to determine how useful

Table 2 Table of Common Metric Notation

α	Reputation accuracy
α_ρ	Random selection accuracy
χ	Reputation convergence
ϵ	Reputation effect
R_σ	Reputation from random selection
R_ρ	Reputation from reputation mechanism
R_β	Actual peer behavior
n	Number of peers available
τ	Threshold for residual convergence

a particular reputation mechanism will be under a given system and preference model. We provide general forms for each of the metrics to enable consistent use of the metrics. The general form allows mechanism developers to assure that the specific form they use to describe their mechanism describes the correct aspects of mechanism behavior.

4.1 Reputation Accuracy

Reputation accuracy describes how accurately the reputation mechanism labels and predicts the actions of other peers (or resources). Accuracy is a value within the continuous range of [0..1]. A high accuracy value does not necessarily imply secure selection of peers since all peers in the system may be faulty or malicious. Rather a high accuracy value implies that the reputation mechanism is able to accurately label the level of trustworthiness of peers. As shown in Section 5, the exact formulation of reputation accuracy depends on the reputation mechanism being evaluated; however, Equation 6 provides the general case for the value.

A reputation mechanism should perform better than random selection for a given attack or it is worthless against that particular attack. Intuitively, we compare to random selection because a reputation mechanism that operates in a system with completely reliable peers could easily provide the same benefit as randomly selecting peers. The normalized accuracy metric shows that the reputation mechanism in that situation would provide no benefit to the system, and incur the computational and communication overheads of running a reputation mechanism. As a result, we also describe a metric for determining the success of the reputation mechanism in comparison to no mechanism at all with Equation 7.

$$\alpha = \frac{\sum_{i=0}^{n-1} |R_\rho^i - R_\beta^i|}{n} \tag{6}$$

$$\alpha' = \frac{\sum_{i=0}^{n-1} |\frac{R_\rho^i - R_\beta^i}{R_\sigma^i - R_\beta^i}|}{n} \tag{7}$$

4.2 Reputation Convergence

Reputation convergence describes how close to the actual convergent reputation value a reputation value will be within a given time frame. Convergence is a value

within the range of [0..1]. For example, in a centralized system such as Ebay [2009], the convergence value will typically be 1 since the values required for the reputation computation are all available in the central peer; however, in a system such as Credence (Walsh and Sirer [2006]), the reputation information is distributed throughout the system on individual peers. The information must be retrieved and cross-correlated which means that connectivity disruptions can cause values to be unreliable at a given time. As the convergence of a distributed mechanism may never reach a final value, we will define convergence to describe the point when the residual change drops below a threshold τ. That is to say a mechanism is τ-convergent when the change consistently drops below $\tau\%$ of the previous value. As with reputation accuracy, the computation of reputation convergence is dependent on the individual reputation mechanism. The general form for reputation convergence can be found in Equation 8. We normalize the residual at a given time with the actual value expected by the reputation computation if perfect information was available ($R_\rho(\infty)$).

$$\chi(t) = \frac{|R_\rho(t) - R_\rho(\infty)|}{R_\rho(\infty)} \tag{8}$$

4.3 Effectiveness

While the previous two metrics can expose the performance of the critical components of a reputation mechanism, they do not reveal the effectiveness of a reputation mechanism for a given system. For example, if the computation of reputation is perfect, but the required information never arrives, then the mechanism provides no benefit. Likewise, if the information is always available, but the computation component does not perform better than random guessing, then the reputation mechanism provides no benefit.

Based on the two previous metrics, we propose a metric of the effectiveness of a reputation mechanism as follows:

$$\epsilon = \frac{\alpha \times \chi}{\alpha_\sigma} \tag{9}$$

Where ϵ is the effect a reputation mechanism has, α is how accurately a reputation mechanism labels peers, and χ is how quickly the reputation value converges (necessary for distributed computation).

The reputation effect gives a quantitative means to analytically compare the effectiveness of reputation mechanisms where no such metric previously existed. Section 5 provides example of how reputation effect can be derived for EigenTrust.

4.4 Utility Modeling

Reputation accuracy, convergence, and effect are useful for comparisons, but they do not provide a complete picture of a reputation mechanism. In order to provide a more comprehensive view of the reputation mechanism, we use utility modeling. Through utility modeling, we can not only incorporate the effect of a reputation mechanism into quantitative comparisons, but also model and compare the reputation mechanism in different systems with different peer preferences. For example, in a file sharing system peer preference may be for bandwidth;

however, in a distributed computation system, a greater preference may be put on the validity of resource access. As a result, rather than saying that a particular reputation mechanism fits a scenario better than another reputation mechanism because of some qualitative properties, we can show that it fits that situation better quantitatively.

Our utility model of P2P systems comes from the general model described in Lagesse and Kumar [2008]. We provide a concise presentation of that utility model for reference in this paper in Equations 10 - 11. A more comprehensive description can be found in Appendix A.

Table 3 Concise Table of Utility Terms

U_{ben}	Utility Model for a Benign Peer
U_{mal}	Utility Model for a Malicious Peer
B_{ben}	Benign Benefit
B_{acc}	Access Benefit
B_{mech}	Mechanism Benefit
B_{mal}	Malicious Benefit
C_{ben}	Benign Cost
C_{mal}	Malicious Cost
C_{vic}	Cost from being a Victim
C_{mech}	Mechanism Cost
C_{disc}	Cost of being Discovered as Attacker
C_{rep}	Overhead cost of reputation mechanism
A_{tot}	Number of peers available
A_{rep}	Number of peers available after applying reputation

$$U_{ben} = B_{acc} - (C_{ben} + C_{vic}) + B_{mech} - C_{mech} \tag{10}$$

$$U_{mal} = B_{mal} - (C_{ben} + C_{disc} + C_{mal}) - C_{mech} \tag{11}$$

Since in a comparison between reputation mechanisms the expected value of the utility from the system itself (such as cost of entering the system, cost of staying in the system, etc.) will be equal for all mechanisms, this section will only focus only on the utility contribution of the reputation mechanism itself. A successful reputation mechanism has four main effects:

1. decreased expected cost of being a victim
2. decreased expected benefit of a successful attack
3. reduction of available resources
4. increased overhead cost

The first two are similar in that they are the values from the general utility equation multiplied by the reputation effect. The third factor is the result of error in the reputation mechanism that causes a peer to incorrectly refuse

k	Number of benign peers
n	Total number of peers
E	Number of unique peers previously interacted with
B	Attacking peer class
D	Collusive peer class
P	Average number of requests between benign peers
Q	Probability of success with a benign peer
t	Current timestep since start of system
f	Attack or failure rate
r	Rate of interaction of malicious peers
$RepR$	Reported reputation value
$ActR$	Actual reputation value

resources from a benign peer. The fourth factor is the overhead cost for using the mechanism, expressed as a normalized form of the service, processing, memory, and communications costs added by the reputation mechanism, weighted based on the preferences of the peer.

The utility equations for a benign peer and a malicious peer with the addition of a reputation mechanism in the system can be seen in Equation 12 and 13 respectively, where A_{rep} is the amount of a desired resources available after the reputation mechanism is applied and A_{tot} is the amount that would have been available without the reputation mechanism.

$$U_{ben} = B_{ben} \times \frac{A_{rep}}{A_{tot}} - C_{ben} - C_{vic} \times \epsilon - C_{rep} \tag{12}$$

$$U_{mal} = B_{mal} \times \epsilon - C_{mal} - C_{ben} - C_{rep} \tag{13}$$

5 Case Study

In this section we present a case study of the EigenTrust reputation mechanism presented by Kamvar et al. [2003] using the reputation metrics described in Section 4.

5.1 EigenTrust

EigenTrust is a reputation mechanism that uses transitive trust to determine a global trust value for each peer. The mechanism populates a matrix of the number of positive interactions minus the number of negative interactions with each peer (where any negative value is replaced with a 0). The matrix is then normalized so that each row sums to 1. Then the eigenvector of the matrix produces a column vector that contains the global reputation value for each peer. EigenTrust selects peers with the global reputation as the probability of selection. The completely distributed version of EigenTrust performs this computation by exchanging reputation information with peers that have interacted together. The resulting reputation values are then stored in a distributed hash table to reduce the likelihood of successful manipuation of reputation values.

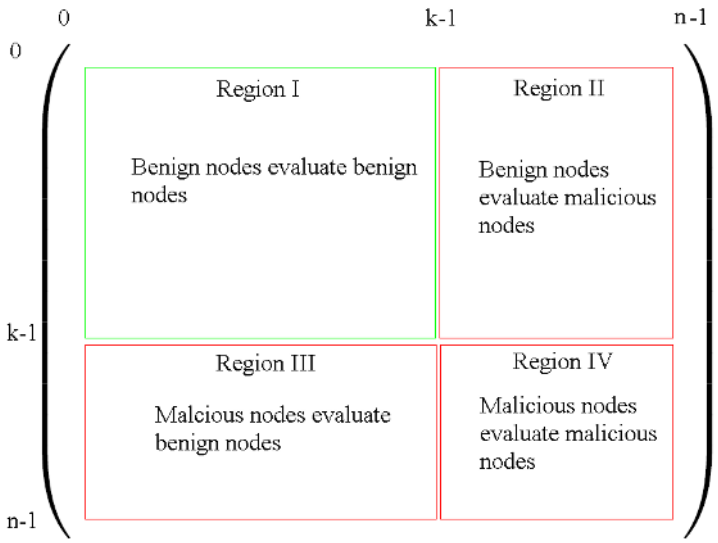

Figure 2 EigenTrust Matrix

5.2 Analysis

We begin by analyzing the reputation accuracy and reputation convergence metrics. Since reputation is considered transitive in EigenTrust, the effect of any given peer's opinion on the calculation of the global reputation value is equal to the difference from the real value times the reputation that the peer has obtained, as shown in Equation 14. The rate of convergence of the reputation value in EigenTrust can be described by the rate of convergence of the distributed eigenvector calculation. This is shown in Equation 15.

$$\alpha = \sum R_{\rho}^{i} \times |RepR_{i,j} - ActR_{i,j}| \tag{14}$$

$$\chi = \frac{E}{N^{N-1}} \times \frac{\lambda_2}{\lambda_1} \tag{15}$$

Where the reputation accuracy, α, is computed with R_{ρ}^{i} as the global reputation value for peer i which is the i^{th} element of the principal eigenvector of matrix R. $RepR_{i,j}$ is the reported reputation of j by i and $ActR_{i,j}$ is the actual reputation that should have been reported for j by i.

The reputation convergence, χ, is computed with E being the number of unique peers previously interacted with (included pre-trusted peers) and n peers in the system to compute the average local network size per peer and λ_2 and λ_1 are the second and first eigenvalues of the normalized reputation matrix (since the matrix is a transition matrix, λ_1 will always be 1).

In the discussion that follows, all parameters used follow the simulation setup described in Kamvar et al. [2003] unless otherwise noted. In the case that we were

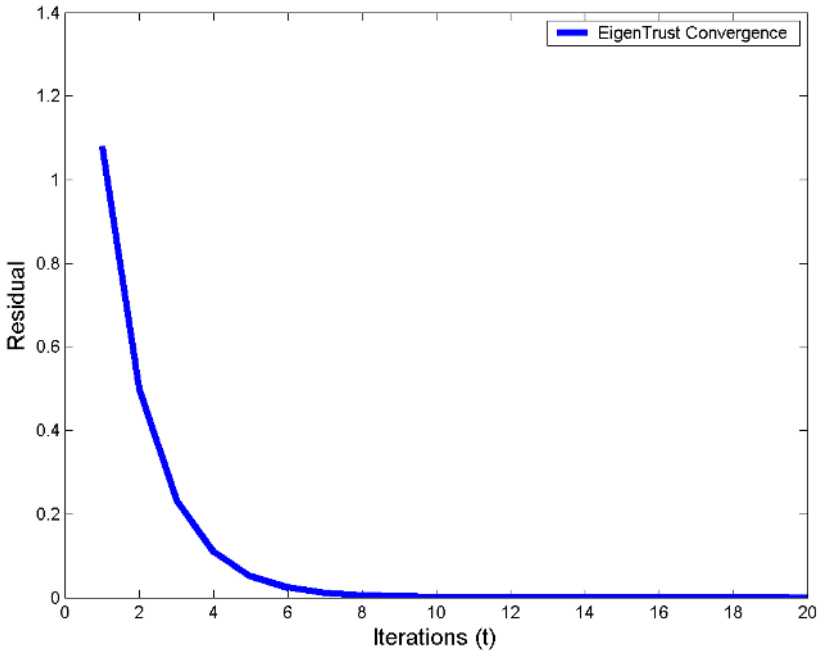

Figure 3 EigenTrust Remaining Residual

uncertain about a parameter needed for our calculations, we have noted our estimate of that value.

In order to compute the residual as shown in the EigenTrust paper, we take the matrix convergence and raise it to increasingly larger powers to calculate how much of the residual remains after each exchange of reputation information as shown in Figure 3. In comparing our analytical results to the simulation results of EigenTrust, we note that we have estimated an average coverage of 20% of the system and a second eigenvalue of 0.465. Our results are similar to those from simulation data, though the simulation shows a slightly quicker convergence than our analytical results.

The accuracy of EigenTrust is dependent on the amount of reputation a peer can obtain times the amount it can deviate its opinion from the truth. Hence, the goal of an attack (particularly a collusive attack with $k - n$ attackers as based on Figure 2) against an EigenTrust reputation mechanism is to maximize the average value of Region II from Figure 2 (which describes an EigenTrust matrix with n peers and k benign peers) in order to maximize the effect of Regions III and IV (which are easily controllable by an attacker). Likewise, the reputation value will converge largely based on the amount of interaction in the system along with the convergence rate of the eigenvector, as noted in Haveliwala and Kamvar [2003]. If each peer in the system has interacted with every other peer in the system, then the system is fully connected and $E = N \times N - 1$, resulting in the only factor of convergence being the second eigenvalue of the reputation matrix. As a result,

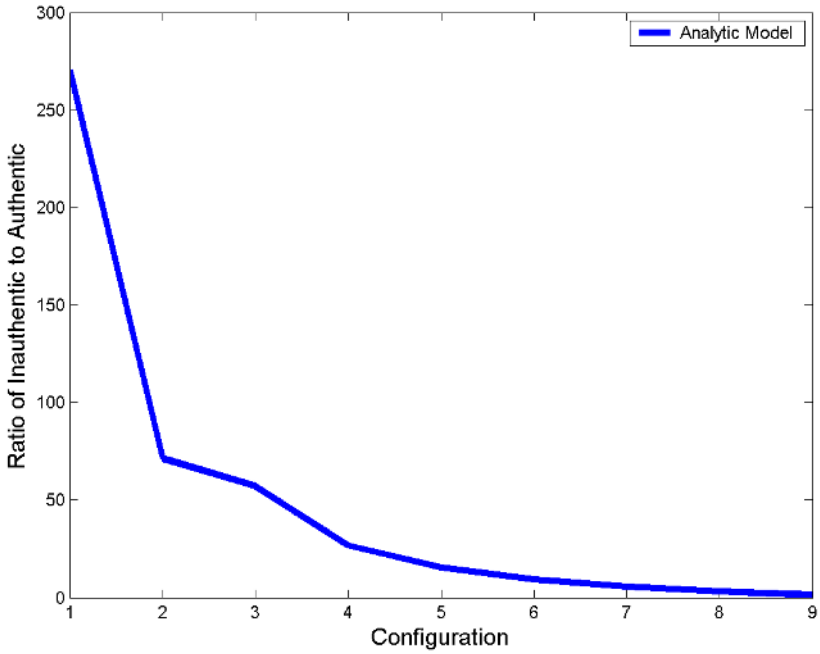

Figure 4 Malicious Spy Attack

we see that the reputation effect from EigenTrust would degrade when applied to sparsely interacting systems.

We can analytically produce results that approximately match the simulations done by Kamvar et al. [2003]. Since the attack of a naive individual is not an interesting case in reputation mechanisms, we will focus our discussion on collusive attack strategies. In all cases, Region I will always be populated (before normalization) with the average value of $P(t) \times (2 \times Q - 1)$ where P is the average number of requests by benign peers to benign peers in the system up through time t and Q is the average quality of benign peer uploads (this value is set to 0.95 to account for the fact that benign peers sometimes make mistakes). The reason for the term $2 \times (Q - 1)$ is that for every negative transaction, the peer will have its score reduced, rather than just not increased. This function produces the expected rating of a benign peer by another benign peer at a given time. In the collusive attacker models, attackers set benign peer ratings to 0, so Region III will be all be 0, and attackers set each others' reputation to 1.

The most interesting region for studying collusive attacks is Region II. This is the region that defines to what extent benign peers trust malicious peers. The first collusive attack simulated in EigenTrust involves all malicious peers always attacking. As a result Region II will become 0 and effectively render the attackers useless when the eigenvector is calculated and as a result we can analytically produce approximately the same results as the EigenTrust simulations. The second attack involves attacking at a rate of $f\%$. Analytical evaluation of this attack

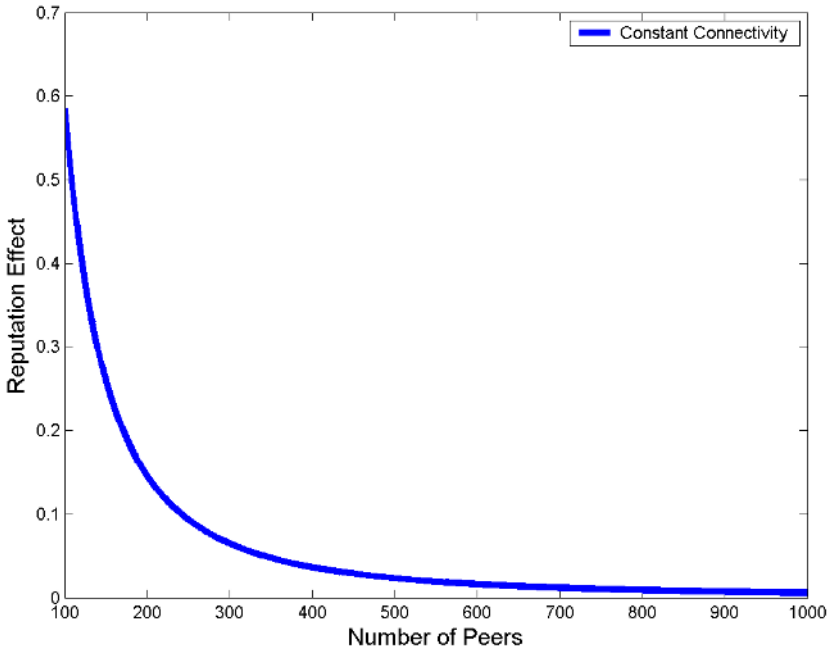

Figure 5 The Effects of Sparse Connectivity on EigenTrust

becomes slightly more difficult because of the probabilistic component; however, we can sum the expected reputation of the average peer in Region II over the course of the system run where the expected reputation at time t is defined by Equation 16 where i is the current reputation of a peer, f is the rate at which the malcious peer attacks, and r is the rate at which malicious peers interact.

$$ExpectedTrust(t) = \sum i \times Pr(i,t) \tag{16}$$

$$\begin{aligned} Pr(i,t) = Pr(i-1,t-1) \times (1-f) \times r + Pr(i,t-1) \\ \times(1-r) + Pr(i+1,t-1) \times f \times r \end{aligned} \tag{17}$$

In Figure 4 we present the results of the most effective attack against EigenTrust, in which peers divide the labor of obtaining good reputations and attack. To do this we identify two classes of malicious peers, D and B. Peers of class D obtain high reputations and report that the attackers, class B, are highly reputable. In order to validate the model, we recreated similar results by solving for the eigenvector of the reputation matrix in order to provide a steady state analysis (which is what EigenTrust simulations do by dropping the results from the first 15 query cycles of each simulation).

In Figure 5 we show the effect of increasing the number of peers in an EigenTrust system while holding constant the average rate of connectivity between peers. For

this figure we start with a λ_2 value of 0.65, a fully connected system ($E = N \times (N-1) = 9900$), and a perfectly accurate reputation report ($\alpha = 1$). As a result, it is obvious that EigenTrust is effective in reasonably well-connected environments; however, it quickly degrades in quality with an increase in the number of information exchanges needed for the global reputation values to converge.

It is noted that the point of this exercise is to show that a reputation mechanism can be broken down analytically. As the entire details of every simulation result in Kamvar et al. [2003] were not available, we made some assumptions to compute results. In general the pattern of the plots in Figures 3, 4, and 5 are similar to those in Kamvar et al. [2003]. Through the proposed model, developers and researchers can compare and evaluate their reputation mechanisms, provided the appropriate analytical equations and algorithmic descriptions are available.

6 Conclusion

In this paper we have presented several metrics and a utility-based approach for the analytical evaluation of reputation mechanisms in P2P systems. We believe that these approaches provide a tool for evaluating these mechanisms where previous evaluations fell short. The analytical evaluation of reputation mechanisms allows a system designer to compare the performance of a variety of reputation mechanisms under varying circumstances without the need for writing simulations. As a result, it becomes easier to compare and share results with others. Further, the mathematical breakdown of reputation mechanisms can assist in discovering the root cause of weaknesses in the mechanism as shown in our case study in Section 5.

Acknowledgements

Prepared by Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6285, managed by UTBattelle, LLC, for the U.S. Department of Energy under contract DE-AC05-00OR22725.

References

K. Aberer and Z. Despotovic. Managing trust in a peer-2-peer information system. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 310–317, New York, NY, USA, 2001. ACM. ISBN 1-58113-436-3.

C. Dellarocas. Reputation mechanisms. In *Handbook on Economics and Information Systems*, page 2006. Elsevier Publishing, 2006.

Ebay. Ebay, July 2009. `http://www.ebay.com`.

T. Haveliwala and S. Kamvar. The second eigenvalue of the google matrix. Technical Report 20, Stanford University, 2003. URL `http://www.stanford.edu/ taherh/papers/secondeigenvalue.pdf`.

K. Hoffman, D. Zage, and C. Nita-Rotaru. A survey of attack and defense techniques for reputation systems. *ACM Comput. Surv.*, 42(1):1–31, 2009. ISSN 0360-0300.

S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in P2P networks. In *WWW*, pages 640–651, 2003.

B. Lagesse and M. Kumar. A novel utility and game-theoretic based security mechanism for mobile p2p systems. *Pervasive Computing and Communications, IEEE International Conference on*, 0:486–491, 2008.

B. Lagesse, M. Kumar, J. M. Paluska, and M. Wright. Dtt: A distributed trust toolkit for pervasive systems. *Pervasive Computing and Communications, IEEE International Conference on*, 0:1–8, 2009.

J. Mundinger and J.-Y. Le Boudec. Analysis of a reputation system for mobile ad-hoc networks with liars. *Perform. Eval.*, 65(3-4):212–226, 2008. ISSN 0166-5316.

A. Nandi, T.-W. J. Ngan, A. Singh, P. Druschel, and D. S. Wallach. Scrivener: providing incentives in cooperative content distribution systems. In *Middleware '05: Proceedings of the ACM/IFIP/USENIX 2005 International Conference on Middleware*, pages 270–291, New York, NY, USA, 2005. Springer-Verlag New York, Inc.

M. Srivatsa, L. Xiong, and L. Liu. Trustguard: countering vulnerabilities in reputation management for decentralized overlay networks. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 422–431, New York, NY, USA, 2005. ACM. ISBN 1-59593-046-9.

K. Walsh and E. G. Sirer. Experience with an object reputation system for peer-to-peer filesharing. In *NSDI*. USENIX, 2006.

A Utilty Model

Table 4 Table of Utility Terms

U_{ben}	Utility Model for a Benign Peer
U_{mal}	Utility Model for a Malicious Peer
U_{hyb}	Utility Model for a Hybrid Peer
U	Total Utility
B	Total Benefit
C	Total Cost
B_{ben}	Benign Benefit
B_{acc}	Access Benefit
B_{mech}	Mechanism Benefit
B_{mal}	Malicious Benefit
B_s	Benefit from Spying
B_d	Benefit from Denying Service
B_f	Benefit from Serving Faulty Resources
C_{ben}	Benign Cost
C_{mal}	Malicious Cost
C_{vic}	Cost from being a Victim
C_s	Cost from being Spied On
C_d	Cost from being Denied Service
C_f	Cost from being Served Faulty Resources
C_{conn}	Cost being Connected to the System
C_{res}	Cost of Providing Resources
C_{mech}	Mechanism Cost
C_{ms}	Cost of Spying
C_{md}	Cost of Denying Service
C_{mf}	Cost of Serving Faulty Resources
C_{disc}	Cost of being Discovered as Attacker

$$C_{vic} = C_s + C_d + C_f \tag{18}$$

$$B_{ben} = B_{acc} + B_{mech} \tag{19}$$

$$B_{mal} = B_s + B_d + B_f \tag{20}$$

$$C_{ben} = C_{conn} + C_{res} + C_{mech} \tag{21}$$

$$C_{mal} = C_{ms} + C_{md} + C_{mf} \tag{22}$$

$$B = B_{ben} + B_{mal} \tag{23}$$

$$C = C_{ben} + C_{mal} + C_{vic} + C_{disc} \tag{24}$$

$$U = B - C \tag{25}$$

$$U_{hyb} = (B_{acc} + B_{mal}) - (C_{ben} + C_{vic} + C_{disc} + C_{mal}) \quad (26)$$

$$U_{ben} = B_{acc} - (C_{ben} + C_{vic}) \quad (27)$$

$$U_{mal} = B_{mal} - (C_{ben} + C_{disc} + C_{mal}) \quad (28)$$