Reputation Management: Experiments on the Robustness of ROCQ*

Anurag Garg Roberto Battiti Roberto Cascella
Dipartimento di Informatica e Telecomunicazioni,
Università di Trento,
Via Sommarive 14, 38050 Povo (TN), Italy.
{*garo,battiti,cascella*}*@dit.unitn.it*

Abstract

In order for autonomic systems to function, the individual components must co-operate and not indulge in malicious behavior. However, it is almost certain that autonomous systems in Next Generation Networks will inadvertently include less than trustworthy components. Identifying such entities is critical to the smooth and effective functioning. We present new experiments conducted with the ROCQ scheme, a reputation-based trust management system that computes the trustworthiness of peers on the basis of transaction-based feedback. The ROCQ model combines four parameters: Reputation (R) or a peer's global trust rating, Opinion (O) formed by a peer's first-hand interactions, Credibility (C) of a reporting peer and Quality (Q) or the confidence a reporting peer puts on the feedback it provides. In this paper, we demonstrate that ROCQ is robust against churn and also examine the effect of credibility and quality on the performance of the scheme.

1. Introduction

Autonomous systems attempt to monitor their own behavior (self-aware) and react to malfunctioning components or modules in an automated fashion (self-healing). This paper explores ideas of trust management that have been developed in other contexts such as e-commerce and peer-to-peer systems and applies them for the soft enforcement of rules of behavior in specific autonomous systems.

A substantial body of work has emerged in recent years on trust management in electronic communities. The fundamental concept behind most of this work is to study the past interactions of a user and estimate his or her trustworthiness on this basis. These systems allow users to rate transactions they have had with other users. These ratings are then aggregated to form a global trust metric which is used to calculate the trustworthiness of a user.

Initial trust management systems were based on centralized trust databases. The eBay rating system, the Amazon customer review system and the Slashdot self-moderation of posts [1] are all systems where the ratings are provided by peers but are stored in a central database. Many such reputation systems have been studied in the context of online communities and marketplaces [2–4].

More recently, schemes with decentralized trust databases have emerged, particularly in the context of peer-to-peer networks. Examples of such systems include Aberer et al.'s scheme based on P-Grid [5], Cornelli et al.'s mechanism built on Gnutella [6, 7], the EigenTrust system proposed by Kamvar et. al. [8], PeerTrust by Xiong et. al. [9].

In this paper we examine the ROCQ scheme (pronounced "rock") that computes peer Reputations (R) on the basis of Opinion (O), Credibility (C) and Quality (Q). This scheme was introduced in [10], where results from initial experiments that studied the feasibility and effectiveness of ROCQ were presented. In this work, we extend these results by studying new scenarios. In particular, we study the impact of churn – the continuous process of node arrivals and departures – and of credibility and quality on ROCQ.

The ROCQ model is summarized in Section 2 followed by a description of the system architecture that builds on top of a structured P2P network to provide decentralized trust storage, aggregation and dissemination. Also in Section 3 we discuss security considerations for ROCQ. In Section 4 we describe a series of simulation-based experiments designed to evaluate the performance of ROCQ in a variety of settings.

*This work was supported by the Provincia Autonoma di Trento through the WILMA Project.

0-7803-8963-8/05/$20.00 ©2005 IEEE

2. The ROCQ Model

2.1. Opinion

A peer forms an opinion about the amount of satisfaction it has derived from a transaction that it takes part in. The term O_{ij}^{k} refers to peer i's opinion about its k^{th} transaction with peer j and is normalized to take values in $[0,1]$. A peer may also choose to keep a record of its own first-hand experiences in the form of averaged opinions.

Hence, O_{ij}^{avg} is peer i's estimate of the average amount of satisfaction it has received from peer j and is computed as follows:

$$O_{ij}^{avg} = \frac{\sum_{k=1}^{N_{ij}} O_{ij}^{k}}{N_{ij}} \tag{1}$$

where N_{ij} is the number of interactions it has had with j.

2.2. Reputation

The reputation of a peer i is the end result of aggregating feedback about i from several other peers. It represents the global system-wide view of the average amount of satisfaction a peer is likely to derive through an interaction with i. The reputation of a peer is also normalized so that it lies between 0 and 1.

The reputation of a peer j is computed at the peer m using reported opinions, the quality value sent by reporting peers and the credibility of reporting peers as follows:

$$R_{mj} = \frac{\sum_i O_{ij}^{avg} \cdot C_{mi} \cdot Q_{ij}}{\sum_i C_{mi} \cdot Q_{ij}} \tag{2}$$

where R_{mj} is the aggregated reputation of peer j, C_{mi} is the credibility of peer i according to peer m, O_{ij}^{avg} is the average opinion of j reported by i and Q_{ij} is the associated quality value reported by i. Thus peer m gives more weight to ratings that are considered to be of a high quality and that come from peers who are more credible in its eyes.

2.3. Credibility

In ROCQ, the credibility of a peer is used to weigh the feedback it reports. If a peer gives wrong feedback about other peers its credibility rating is decreased and its subsequent reports have a reduced impact on the reputation of another peer. Similarly, if a peer's feedback is consistently good, i.e., in agreement with other reporting peers, its credibility rating goes up. Credibility ratings are based on first-hand experience only and, unlike opinions, they are not shared with other peers. Credibility ratings are normalized so that they lie between 0 and 1.

When a peer reports an opinion to another peer for the first time, its credibility is set to 0.5. Thereafter, on the $k+1^{th}$ report to peer m, the credibility of peer i, C_{mi}^{k+1} is given by

$$C_{mi}^{k} + \frac{(1 - C_{mi}^{k})}{2} \cdot Q_{ij} \cdot (1 - \frac{|R_{mj} - O_{ij}^{avg}|}{s_{mj}}) \tag{3}$$

if $|R_{mj} - O_{ij}^{avg}| < s_{mj}$ and by

$$C_{mi}^{k} - \frac{C_{mi}^{k}}{2} \cdot Q_{ij} \cdot (1 - \frac{s_{mj}}{|R_{mj} - O_{ij}^{avg}|}) \tag{4}$$

otherwise. C_{mi}^{k} is the credibility of peer i after k reports to peer m, O_{ij}^{avg} is the opinion being currently reported by peer i, Q_{ij} is the associated quality value, R_{mj} is the aggregated reputation value that peer m computed for j and s_{mj} is the standard deviation of all the reported opinions about peer j.

2.4. Quality

ROCQ allows a peer to determine the confidence of its feedback. Giving incorrect feedback can decrease the credibility of a peer. So, a peer can lower the quality value for opinions about which it is not very sure, therefore risking less loss of credibility in case its judgment is incorrect. Quality is also normalized to lie between 0 and 1.

We assume that the amount of satisfaction provided during each transaction by peer j is a normally distributed random variable. Through interactions with j, peer i makes observations of this random variable resulting in a sample. The sample mean and standard deviation are then O_{ij}^{avg} and s_{ij}.

The quality value of the opinion (Q_{ij}) is defined as the confidence level that the actual mean trust rating for a peer lies within the confidence interval:

$$O_{ij}^{avg} \cdot (1 \pm \frac{r}{100}) \tag{5}$$

where r is a system parameter that denotes the size of the confidence interval as a percentage of the sample mean. Further details on how quality values are computed can be found in [10].

3. System Architecture

While the ROCQ trust model is independent of the underlying architecture, its effectiveness clearly de-

pends on the system architecture and the ROCQ implementation. Since there is no centralized database, any implementation needs to collect, store and disseminate reputation information in a distributed way. Moreover, this should be done in a scalable, efficient and cost-effective manner.

Our implementation of ROCQ assumes a structured overlay network that provides a secure, deterministic and reliable way to route messages. These networks use Distributed Hash Tables (DHT) to map objects to a keyspace. Nodes in the network are then responsible for certain ranges of the keyspace. The underlying DHT overlay structure randomly and uniformly designates M **score managers** for each peer in the network. A score manager for a peer is another peer in the network that stores all trust information related to that peer. All feedback pertaining to that peer and requests for the reputation of that peer are routed to the score manager.

When a peer wishes to interact with another peer, it retrieves the reputation values for that peer from its score managers. These are then aggregated using the furnished quality values and the credibility values for the score managers since a score manager itself may be malicious and send the wrong reputation values. The final average reputation value – formed by two aggregations, first at the score managers and second at the requesting peer – is then used to decide whether the transaction should proceed. It is computed as follows:

$$R_{ij}^{avg} = \frac{\sum_y R_{yj} \cdot C_{iy} \cdot Q_{yj}}{\sum_y C_{iy} \cdot Q_{yj}} \tag{6}$$

where R_{yj} is a reputation value received from a score manager y about peer j.

If a peer has had interactions with the prospective partner before, it may have already formed an opinion value for this peer. In this case, the peer may wish to prefer its own first-hand experience to the information being provided by the trust management system or to use a combination of the global reputation and its first hand experience.

3.1. Security Considerations

Let us now briefly discuss the security problems that arise in the context of ROCQ. The ROCQ reputation mechanism cannot prevent all kinds of attacks that malicious entities can inflict on autonomous system. ROCQ is a soft mechanism that attempts to enforce good behavior in the network through incentives. Hence attacks on the integrity of the underlying network, such as misrouting messages or a single user presenting multiple identities to subvert the system are out of the scope of ROCQ. We describe some of these attacks and solutions that have been presented in the literature.

Autonomous systems usually create an overlay network for communication between the constituents. These overlay networks require collaboration among users to forward messages and to update routing tables in a consistent manner. Malicious entities can disseminate false routing updates in the network and they can falsify information about the location or status of a resource as so to keep control on the resource.

However, failure in message delivery can be caused either by topology changes due to the dynamic nature of such networks or by a compromised node. As a result, it is not trivial to distinguish between benign and malign failures. And because the operability of such networks depends on the nodes' willingness to forward messages, a compromised node will affect more than just the interactions it is part of. Failure tests or redundant [11] and iterative routing [12] can be used to detect faults and to provide alternative routes around the failed node. But this results in a cost increase since multiple paths need to be stored and updated.

Other possible attacks on autonomous systems include denying the existence of data the node is responsible for and repudiating transactions. The first type of attack can be avoided by carefully replicating the object at different locations not under the control of the same node. This approach can be beneficial when the data stored can be tampered or mislabeled. However, such a replication scheme will not be effective when a node can assume multiple identities. This attack, known as Sybil attack [13], can severely compromise an autonomous network, since malicious nodes can counterfeit identities with different reputation values and control different object in the system. This eliminates any benefit that could have been obtained from a reputation management scheme, from replicating data or from using alternative paths for routing. In [13], Douceur suggests using a trusted identification authority that is in charge of establishing node identifiers. In [12] certified nodeIDs are proposed; this requires each entities to own a certificate (valid public/private keys) and binds the nodeID to a specific IP address. This has several drawbacks since node mobility or changes in the network can cause the node's IP address to change requiring re-certification or the assumption of a new identity. Furthermore, the solution does not work for nodes behind a NAT.

The presence of a certification authority allows a public key infrastructure to function, which in turn allows for encryption of messages and signing of feedback messages. This ensures the integrity of trust query, reply and feedback messages in transit. Furthermore,

a peer cannot repudiate a message once it has been sent and recipients are able to verify the identity of the sender and the authenticity of the message.

4. Experimental Results

Preliminary experimental results of ROCQ were presented in [10]. The new results we present below focus on the performance of ROCQ under churn and the impact of quality and credibility.

4.1. Methodology

For our experiments, we use FreePastry [14], an open-source implementation of Pastry that is written in Java. We used FreePastry to create a virtual P2P network and to deliver messages between peers. ROCQ is implemented as an application that runs on top of individual Pastry nodes.

Number of Peers and Interactions. Unless stated otherwise, each experiment simulates a network with 200 peers where 50000 transactions take place. Both participants of each interaction are chosen randomly. The default number of score managers storing reputation ratings for each peer is 6. Each experiment was performed 10 times and the average of the results is plotted, along with a confidence interval of size $\frac{s}{\sqrt{10}}$ where s is the standard deviation.

Performance Metric. The performance is measured as the number of correct decisions made (i.e., interactions with good peers that went ahead plus interactions with malicious peers that were avoided) as a proportion of the total number of decisions made. Only decisions made by good peers were counted.

Type of Maliciousness. We simulate two different kinds of maliciousness. A peer can be malicious in the base system, i.e., behave maliciously when interacting with other peers and/or it may be malicious in the reputation system sending incorrect reputation values to requesting peers.

4.2. Comparison with the Aberer-Despotovic Scheme

In this experiment we compare the performance of the ROCQ scheme with the trust management scheme proposed by Aberer and Despotovic in [5].

Aberer and Despotovic proposed two schemes, the *simple* and the *complex*. In the simple scheme, peers give equal weight to all reporting peers. In the complex scheme, only reports from peers that exceed a given trust threshold are taken into account.

Figure 1 compares the performance of the two

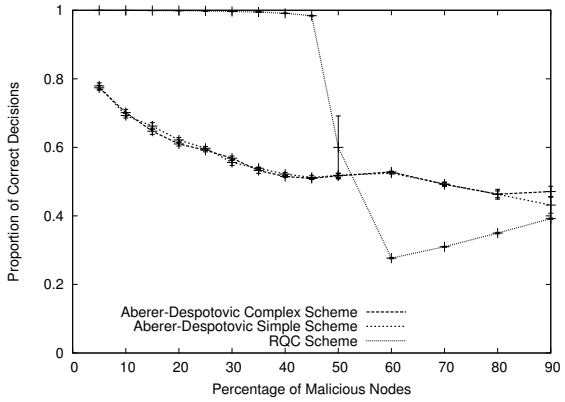

Figure 1. Comparison of the RQC Scheme with the Aberer-Despotovic Scheme with Malicious in the Base System Only

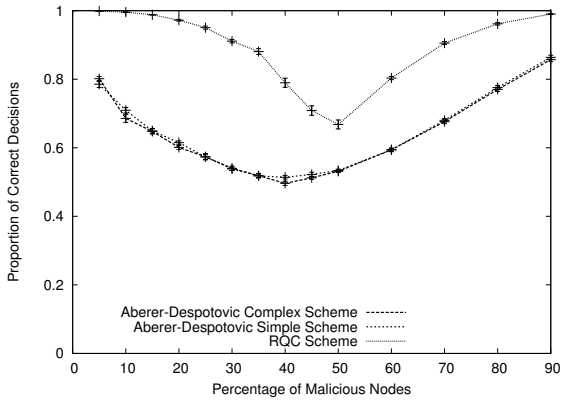

Figure 2. Comparison of the RQC Scheme with the Aberer-Despotovic Scheme with Maliciousness in both Base and Reputation Systems

schemes when there is maliciousness in the base system only whereas Fig. 2 compares the performance when there is maliciousness in both the base and the reputation system. ROCQ outperforms both the simple and complex the Aberer-Despotovic algorithms in all cases except when the proportion of malicious peers is higher than 50% in Fig. 1. In this case, the dominant ethic of the system is that of the malicious peers.

4.3. The Impact of Churn in the Network

In this experiment we measure the impact of churn - the continuous process of peers joining and leaving the network - on the performance of ROCQ. This experiment was performed with 30% malicious peers in the base system only. The straight line indicates the performance of ROCQ with no churn. The x-axis indicates,

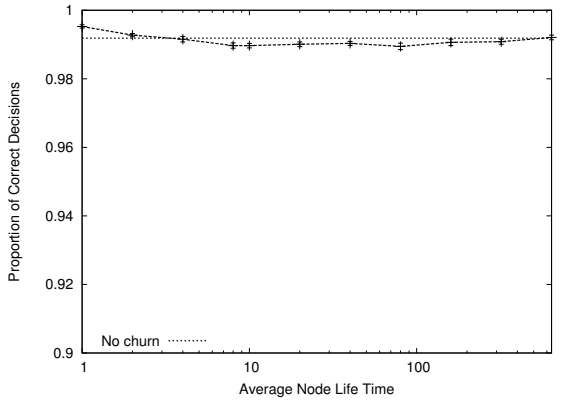

Figure 3. Impact of Churn on ROCQ with Maliciousness in the Base System Only

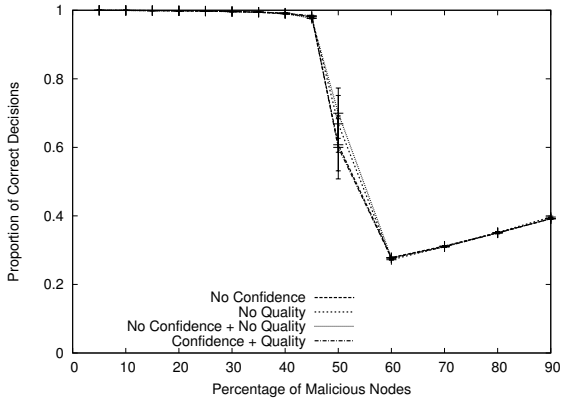

Figure 4. Impact of Credibility and Quality on ROCQ with Maliciousness in the Base System Only

on a logarithmic scale, average peer lifetime in terms of number of transactions the peer participates in. As the average peer life time increases, the amount of churn in the network increases.

Note that an average lifetime of two transactions does not necessarily mean that each peer leaves the network after two transactions. Peers join and leave the network at random and several peers may leave the network without participating in any transaction. On the other hand, several peers may remain in the network for a substantial duration of the experiment. For instance, in our tests with an average peer lifetime of 1.5 transactions, the longest lived peers took part in up to 20 transactions. Moreover, we do not count *initial transactions* – defined as transactions with a peer never seen before – when measuring the performance of ROCQ. When the average peer lifetime is small, the number of initial transactions is considerably higher, thus excluding them from the performance statistics.

As we can see in Fig. 3, ROCQ performance remains unchanged over a wide range of churn rates. From this result, we can infer that the ROCQ trust model and system architecture are very resilient to churn. The system preserves enough redundancy that node departures do not result in a loss of trust information and at the same time, new arrivals are gracefully integrated into the trust management system.

4.4. Impact of Confidence and Quality

In this experiment we turn our attention to the effect quality and credibility have on ROCQ. Recall that the variable credibility represents a peer's honesty in the reputation system whereas quality is the importance a peer attaches to feedback it sends. In this experiment we examine the cases when (1) Both credibility and quality values are used, (2) Only credibility values are used, (3) Only quality values are used and (4) Neither credibility nor quality values are used to compute peer reputations.

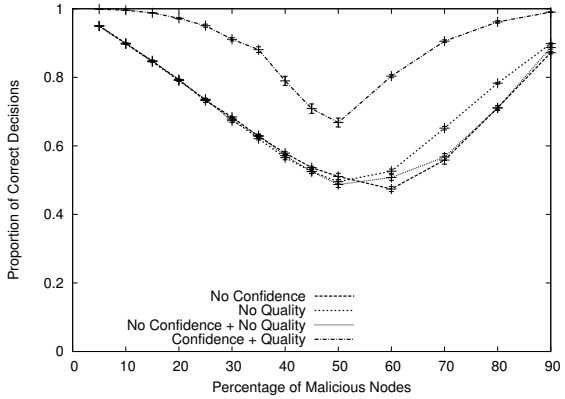

Figure 5. Impact of Credibility and Quality on ROCQ with Maliciousness in the Reputation System Only

Figure 4 shows that furnishing credibility and quality values does not have much impact on the performance of ROCQ when there is maliciousness in the base system only. This is to be expected as when peers are malicious in the base system only, they do not give incorrect recommendations and hence full confidence can be placed in all recommendations.

On the other hand, Fig. 5 shows the performance of ROCQ when there is maliciousness in the reputation system only. Here, the absence of credibility and quality values reduces the performance of ROCQ by up to 30%. Moreover, of the three poorly performing

cases, not using quality values performs marginally better. This shows that credibility is the more important of the two variables. It is clear from Fig. 5 that quality and credibility are critical to the performance of ROCQ and opinion, quality and credibility should all be taken into account to accurately measure the reputation of a peer.

5. Conclusions

In this paper we have presented new experimental results with ROCQ, a reputation-based trust management system. The experiments demonstrate that ROCQ is resilient to node churn over a variety of churn rates. In addition, we examine the impact of credibility and quality on the performance of ROCQ. We introduced the concepts of credibility and quality in [10] as part of the ROCQ algorithm. We now demonstrate that these variables add value to the ROCQ algorithm and without them ROCQ performance decreases by up to 30%. Finally, we discuss the security requirements and implications of a scheme like ROCQ. Preventing certain kinds of attacks, especially those that target the integrity of the underlying network, is beyond the scope of ROCQ.

References

[1] Cliff Lampe and Paul Resnick. Slash(dot) and burn: distributed moderation in a large online conversation space. In *Proceedings of the 2004 conference on Human factors in computing systems*, pages 543–550. ACM Press, 2004.

[2] Paul Resnick, Richard Zeckhauser, John Swanson, and Kate Lockwood. The value of reputation on eBay: A controlled experiment. Working paper originally presented at the ESA conference, June 2002.

[3] Chrysanthos Dellarocas. Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. In *Proceedings of the 2nd ACM conference on Electronic commerce*, pages 150–157. ACM Press, 2000.

[4] Giorgos Zacharia, Alexandros Moukas, and Pattie Maes. Collaborative reputation mechanisms in electronic marketplaces. In *Proceedings of the Thirty-second Annual Hawaii International Conference on System Sciences-Volume 8*, page 8026. IEEE Computer Society, 1999.

[5] Karl Aberer and Zoran Despotovic. Managing trust in a peer-2-peer information system. In *CIKM*, pages 310–317, 2001.

[6] F. Cornelli, E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. Choosing reputable servents in a p2p network. In *Eleventh International World Wide Web Conference*, Honolulu, Hawaii, May 2002.

[7] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. Managing and sharing servents' reputations in p2p systems. *IEEE Transactions on Data and Knowledge Engineering*, 15(4):840–854, Jul./Aug. 2003.

[8] Sepandar D. Kamvar, Mario T. Schlosser, and Hector Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the twelfth international conference on World Wide Web*, pages 640–651. ACM Press, 2003.

[9] Li Xiong and Ling Liu. PeerTrust: Supporting reputation-based trust in peer-to-peer communities. *IEEE Transactions on Data and Knowledge Engineering, Special Issue on Peer-to-Peer Based Data Management*, 16(7):843–857, July 2004.

[10] Anurag Garg, Roberto Battiti, and Gianni Costanzi. Dynamic self-management of autonomic systems: The Reputation, Quality and Credibility (RQC) scheme. In *The 1st IFIP TC6 WG6.6 International Workshop on Autonomic Communication (WAC 2004)*, October 2004.

[11] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. Wallach. Secure routing for structured peer-to-peer overlay networks. In *Fifth Symposium on Operating Systems Design and Implementation (OSDI '02)*, Boston, Massachusetts, December 2002.

[12] E. Sit and R. Morris. Security considerations for peer-to-peer Distributed Hash Tables. In *The 1st International Workshop on Peer-to-Peer Systems (IPTPS 02)*, Cambridge, Massachusetts, 2002.

[13] John Douceur. The Sybil Attack. In *Proceedings of the 1st International Peer To Peer Systems Workshop (IPTPS 2002)*, Cambridge, Massachusetts, March 2002.

[14] Antony Rowstron and Peter Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pages 329–350, November 2001.