The 2nd International Conference on Computer Science and its Applications (CSA 2009), IEEE Publication, Jeju, Korea, December 2009.

Trusted P2P Group Interaction

Rachid Anane[1], Steven Marrocco[2] and Behzad Bordbar[2]

[1]*Faculty of Engineering and Computing, Coventry University*
r.anane@coventry.ac.uk
[2]*School of Computer Science, University of Birmingham*
{s.marrocco, b.bordbar}@cs.bham.ac.uk

Abstract

The issue of trustworthy transactions in P2P systems is often addressed by deploying schemes based on evidence and recommendation. Most implementations rely on a carefully designed formula for expressing and recording the historical performance of peers and for determining the level of trust that should govern the nature of their interactions. In this paper, a more comprehensive approach to trust management is proposed, where trustworthy transactions are supported by first, the historical behaviour of the peers, through formula computation for local trust and distributed trust, second, group management and third, system configuration and parameter tuning. These features are supported by a layered architecture and are designed to provide a more realistic trust evaluation in P2P environments.

1 Introduction

In most client/server architectures an omniscient server is the focal point of activity and the repository of critical information for the whole distributed system. Brokering functions and schemes based on public key infrastructure (PKI) can be easily mapped onto a client/server architecture. The functional asymmetry that distinguishes client/server models from P2P systems allows for the implementation of protection mechanisms such as authorisation and server authentication. In contrast, the symmetric role that peers assume in a P2P system and the arbitrary formation of P2P networks, as well as their evolution, require alternative mechanisms for supporting secure transactions [1]. The absence of a centralised server has two important implications. The first refers to the need for peers to contact directly other peers and to discover by themselves the resources they require. The second involves the setting up of alternative or complementary mechanisms for supporting reliable transaction management.

One of the key characteristics of P2P systems is the ability of peers to cooperate in a decentralised manner to achieve a common goal. The ease with which peers can enter or leave a network underlines their scalability and their resilience. This openness and the relative anonymity of the transactions that can be conducted in a P2P system can be a source of vulnerability to attacks by malicious peers and to abuse by free riders [2]. This concern has led to the deployment of various trust and reputation systems in e-commerce and other environments [3].

Its is the lack of accountability in P2P networks and their decentralised nature precludes the exclusive reliance on security systems such as public key infrastructure (PKI), where the emphasis is on ‘secure’ and ‘certified’ transactions. Although digital signatures can form the basis for authentication as in Kazaa for example, most P2P systems rely on social networks or trust/reputation-based systems [4]. Trust may be defined as the belief of a peer in the reliability and honesty of another peer. This belief is the result of direct interaction between any two peers and is also referred to as local trust [5]. This form of trust, which may involve the exchange of keys, is associated with evidence-based systems. Reputation, on the other hand, is defined as the belief of a peer in the reliability and honesty of another peer, generated from the recommendation or reference from other peers. Reputation is the basis of recommendation-based systems and is also referred to as distributed or aggregated trust [6].

While evidence-based systems are characterised by a limited span of interaction, in recommendation–based systems the scope of interaction may involve a large number of peers. Trust-based systems rely on a formula that combines trust and reputation factors, assign trust levels according to past behaviour and constrain the level of interaction according to threshold values. This focus on the determination of

trust level through formula calculation has however its limitations. Trust is context sensitive, multi-faceted and dynamic and as such it is often difficult to encapsulate these characteristics in a formula [5]. For example, peers may be penalised because of network latency. Reliance on reputation only may not be useful, especially for new peers, and the need to submit queries at large to request information may be intrusive, inefficient and unreliable.

The emergence of P2P systems as a source of valuable information has highlighted the need for effective means of locating relevant and useful resources. In resource discovery, the highly dynamic nature of P2P networks may be incompatible with centralised mechanisms, especially in unstructured P2P networks such as Gnutella. Query propagation and the high level of network activity that this entails are among the most influential factors in the formation of communities of interest [7].

This work is motivated by the need to enhance formula-based trust systems and to address some of the limitations of existing approaches. The support for trustworthy transactions is articulated by the following considerations:

- Trust levels through formula calculation in terms of historical records of peer behaviour and performance can be very effective.
- Structured approaches to interaction, which allow the formation/disbanding of interest groups, can improve the reliability of transactions and the efficiency of query management.
- Hybrid systems, which combine evidence and recommendation to determine trust levels, offer flexibility. A basic PKI scheme can enhance the context of trust and reputation mechanisms, through mutual authentication.
- Awareness that environmental factors such as network latency are beyond the control of peers and can affect adversely their behaviour and their trust level. The ability to configure the trust model and to tune system parameters can lead to a more realistic trust evaluation.

These considerations have been translated into a layered architecture, which addresses the specific issues that arise in P2P transaction management. Each layer performs specific functions and contributes to the creation of an environment where secure and trustworthy transactions can be conducted.

This remainder of the paper is organised as follows. Section 2 covers issues related to trust and group management. Section 3 gives an outline the architecture of the proposed system and of the functionality of the core layers. Section 4 gives a brief discussion of relevant issues and pointers for further work. Section 5 concludes the paper.

2 P2P modes of interaction

The harnessing of the resources mediated by P2P networks is facilitated by the introduction of policies and the implementation of mechanisms for trust management and group management.

2.1 Trust rating

The trust rating assigned to a peer should reflect past interactions and take into account the experience of other peers in the system [8]. The rating is usually made up of two components, the local rating and the distributed rating. The effectiveness of local trust rating systems requires the identification of a universally agreed set of criteria for rating peers. This ensures consistency and facilitates the accumulation of distributed ratings.

The specific method used to evaluate the trustworthiness of a peer involves mainly the reliability of a peer and the quality of the service provided. This can be determined by the time taken to respond to a request, or by the quality of the services and resources they provide. Some systems such as those described in [8] base trust evaluation on the levels of complaints received about a peer. Other systems use a binary rating system where an interaction is rated as 1 for satisfactory, or 0 for unsatisfactory [9]. A more flexible approach relies on a continuous range of values to rate the quality of the interaction. This approach relates more to the way trust is established between people, and allows for different levels of trust to exist [6].

Trust evaluation varies from peer to peer depending on the interactions that took place. Distributed trust or aggregated trust is the result of ratings collected from a set of peers, and is designed to give a fuller picture of the behaviour of a peer. A peer requires this aggregated trust when it wants to interact with a peer for which is has no local trust value, or with which it has had very few interactions. To generate this aggregated value a subset of known peers are asked to provide a reference for a specific peer. In most trust systems, where rating requests are forwarded, each peer in the request chain returns only their local value and does not calculate an aggregated value.

Trust systems make use of a variety of mechanisms for collating the results and for calculating the aggregated trust. In systems such as those proposed in [9] and [10], the aggregated rating is scaled by a credibility factor in order to differentiate between the

reliability and the credibility of a peer. Poor recommendations have therefore less influence on the rating of other peers.

2.2 Communities of interests

Groups can be formed either explicitly or implicitly. In an explicit scheme, peers declare their interests, either from a pre-defined list of interests or from one generated arbitrarily. The peer's interests are then made available to its neighbours and to any peer querying its interests. When other peers with a shared interest become aware of its existence, they inform it of their mutual interest. A simple declaration of the same interest by a set of peers creates implicitly a group where peers share common interests. A peer is usually aware of all the peers in a group, or maintains a list of a subset of the peers in the group, usually the peers with a high rating.

In implicit group formation the interests of a peer are generated implicitly by analysing the queries it initiates, and extracting information about their interests [7]. This method provides a more dynamic and up to date representation of the interests of a peer, as they are usually reflected in their most recent activities. This method may give rise to concerns over privacy and may reduce the control that users have over group membership.

Groups in P2P systems can be unmanaged, where each peer is responsible for maintaining information about the group, finding peers in the group and removing themselves from the group if their interests change. A group can elect one peer to act as a coordinator who is responsible for maintaining the registration of all peers in the group and handling queries on their behalf. This type of coordinated group is well suited to partially centralised network models, where the coordinator is designated as a "super peer".

A more discriminate method is detailed by Condie *et al* [11]. In their scheme only the connections to peers that have a likelihood of communicating with each other again are maintained. This is determined by using a trust value for each peer and only maintaining a set number of connections. If a peer becomes available and has a higher trust value than a peer on the list, it is added to the list and the peer with the lowest trust value is removed.

3 Trusted P2P interaction

This work is motivated by the need to identify architectural elements that will contribute to the creation of an environment favourable to the conduct of trustworthy transactions. A set of requirements have been factored out from different perspectives:

- **Separation of concerns:** a layered approach to design and implementation would support the gradual enhancement of a layer without affecting adjacent layers.
- **Contextual support for transactions:** group management should be provided as support for communities of interests and sharing of quality files.
- **Efficient determination of trust**: trust computation should be performed efficiently either locally or aggregated from other peers.
- **Secure communication**: a hybrid approach to security should be implemented in order to capitalise on the advantages of the web of trust and PKI schemes.
- **Responsiveness to network behaviour**: users should be able to configure various parameters via a user interface and adjust system behaviour to network and contextual conditions.

These requirements and relevant issues are best reconciled and integrated by a layered architecture. This type of architecture offers focus in design and enhancement of functionality.

3.1 Layered architecture

In the hierarchy of layers, the lowest layer is the network and the highest is the information management layer:

- **Information layer:** provides functionality for querying for resources, viewing query results, requesting files returned as the result of queries and setting the parameters for adjusting queries.
- **Group layer**: provides controls for the user to add and remove interests, allows the user to view the peers in a group, to view files made available by a group and to request them.
- **Trust layer:** allows the user to view the local trust rating of any peers they are connected to, to rate files that have been received, and to request the distributed trust value of a peer.
- **Security layer**: presents details about the digital certificate in use, the digital certificate of the server, allows the identity of peers to be confirmed, and informs the user about the state of encrypted communication.
- **Network layer:** controls the connection to the network, communication and monitoring the status of peers

Different technologies have been combined into a coherent system for supporting secure and trustworthy transactions. The system was written in Java and the networking and communications

implemented with TCP sockets over IP. The encryption and the generation of certificates made use of The Bouncy Castle Crypto APIs.

3.2 Network and security management

The system conforms to an unstructured P2P network, with no fixed topology, similar to Gnutella. This type of network offers flexibility, resilience and is more suited to highly dynamic P2P networks. It has also the advantage of accommodating heterogeneity of trust and security. As the focus of this paper is mainly on trust management and group management, only a brief outline is given of the functionality of the other layers.

Network management

A hybrid network was chosen in order to support peer discovery and information discovery mechanisms. It involves a combination of explicit and directory methods [12]. An explicit discovery is used for determining the location of the Entry Point Server and requesting a set of neighbours. Once the peers are aware of other peers in the network, they will contact them directly for information about the state of other peers as in the network discovery model. All forms of communication are purely P2P except for the initial contact.

Security management

A trusted third party or Certificate Authority (CA) forms the basis of the security layer. This service is performed by The Entry Point Server and involves support for authentication and a hybrid mode of encryption. This is especially important for peers new to the network, since a web of trust will initially fail to provide references for new peers, and also to deal effectively with impersonation. The CA will be able to ensure that peers have a unique identity and will provide a level of protection against malicious peers. Peers will not be able to interact with other peers unless they authenticate each other. Entity recognition is seen as a fundamental requirement for security management [13].

The use of a hybrid encryption system provides also a balance between security and efficiency, especially when communications consist very often of a single transaction. With digital certificates a combination of public key and private encryption can be used for encrypting the content of the transactions.

3.3 Trust management

Trustworthiness in the P2P network is expressed predominantly in terms of two components: the reliability of a peer and the quality of service provided by a peer. Reliability of a peer involves taking into account its ability to respond to all the requests and its response time. The quality of service is determined by the rating that the user assigns to any received files. Although the quality of service is the most critical factor, it may be affected by the reliability of a peer. As a result, when calculating the trustworthiness of a peer the average file rating is scaled by its reliability.

Reliability

The reliability of a peer is established by assigning to each peer transaction a score of -1, 0, or 1. A score of -1 is assigned to a transaction if no reply is received. A score of 0 is recorded if a reply is received after a set reply time has elapsed. This indicates that the peer is partly reliable. It may be overloaded or communicating over a very slow connection. A score of 1 is assigned to any transaction when a reply is received within the set reply time. For each transaction rated in this way the total number of transactions with that peer is incremented, and the reliability result added to the cumulative total of its reliability.

Quality of service

For every file received by a peer, the user has the opportunity to rate the quality of the file. The user can assign any rating between -1 and 1. This covers a range of possibilities from a harmful or malicious file to an excellent quality file. It also provides more flexibility than a discrete range of values. The suggested rating guidelines include: -1 (Malicious), -0.5 (Misleading), 0 (No Usefulness), 0.5 (Acceptable Quality), 1 (Excellent Quality). A user can freely interpret these guidelines and select any value across the range; for example, a user may rate a “Good Quality File” as 0.7. For every rating for a file received from a particular peer, that rating is added to the peer’s cumulative quality total and the number of files received from that peer is incremented.

Local trust formula

The formula combines reliability and quality of service to generate trust. More specifically, the information gained from rating transactions as the reliability factor, is used to scale the quality of files as determined by the user. This formula takes into account the behaviour of the peers in order to obtain a more realistic trust value. In the formula the local trust value for a peer is computed as the average of the reliability ratings of all transactions with that peer, the reliability factor denoted by *R*, multiplied by the average of the file quality ratings for that peer.

$$R = \left(\frac{\sum_{i=0}^{NumberOfTransactions-1} reliability_i}{NumberOfTransactions} \right)$$

For a positive value of *R*, trust *T* is computed:

$$T = R \times \left(\frac{\sum_{k=0}^{NumberOfTransactions-1} FileQuality_k}{NumberOfFileRatings} \right)$$

Without this qualification, the weighting would be given to the most common type of transaction, very likely an information request and will give no insight into the quality of the resources held by that peer. Efficiency considerations have determined the way the reliability factor is computed and the values that are stored. It is the cumulative number of ratings for that peer and the cumulative sum of the ratings that are stored. In order to calculate the new reliability factor, the current rating is added to the cumulative sum of all ratings and the cumulative number of ratings is incremented.

Distributed trust

When a peer is relatively unknown to the local peer a distributed trust is required for it. A request for reference is sent to peers who are known and who have a local trust value greater than a user configurable threshold value. The recipients of the request determine their own local trust value for the peer in question and return it to the requesting peer. The average of the received local trust values becomes the distributed trust value for the unknown peer. This value is also cached and will be used again for any further interactions with that unknown peer where distributed trust is required. The cache is kept up to date by discarding stale values.

3.4 Group management

An explicit group formation was implemented to support the propagation of queries across groups of interest. It is a flexible and efficient method which is not very intrusive.

Explicit group formation

A peer must declare explicitly an interest if it wants to join a group. This explicit declaration gives the peer control over which groups it wishes to join and thus avoids the privacy issues implied by the implicit approach. An explicit formation has also the added advantage of reducing the amount of data that must be stored by each peer on its neighbours. This method appears also to be the most adaptable and relevant to a decentralised P2P system.

Group discovery

In its search for peers with the same interests a peer will first check the interests of its neighbours. If any of its neighbours have the interest it is looking for, it will ask that neighbour for a list of all peers in that group. Otherwise, it will send a query to its neighbours to be forwarded on its behalf requesting any peer with a common interest to contact it. Peers in a group maintain a list of all other online peers in that group. Each peer is able to query any member of the group and view all of the resources the group is making available. It is also able to provide a new peer who requests information about an interest group with a full list of all peers in that group.

3.5 Peer-to-Peer interaction

The sequence diagram in Figure 1 shows how one transaction can generate multiple transactions, from an initial query for a file. When a query for a file is received by a peer trust is requested and this trust determines whether the request is satisfied. The query is then forwarded until it reaches its propagation limit. The replies are routed back to the source of the query. The diagram indicates also the interplay between query and trust in dealing with a request.

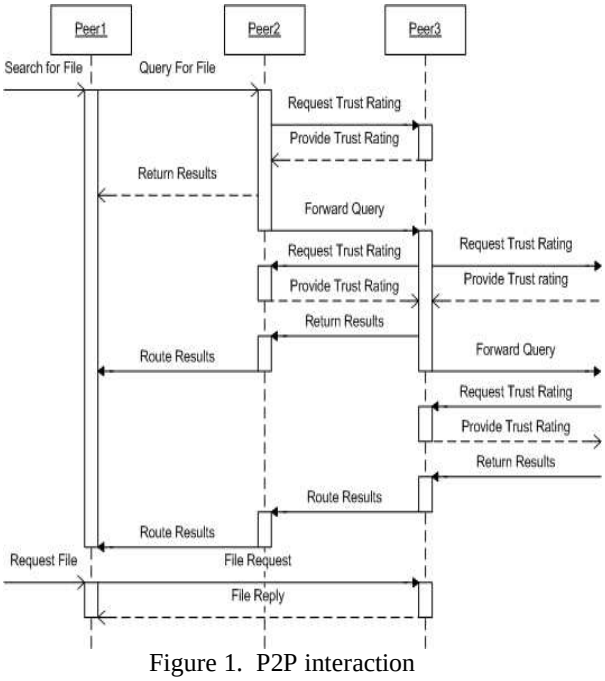

Figure 1. P2P interaction

3.6 System configuration

One major feature of adaptation to context includes the ability of the user to customise system parameters through the Options user interface in Figure 2.

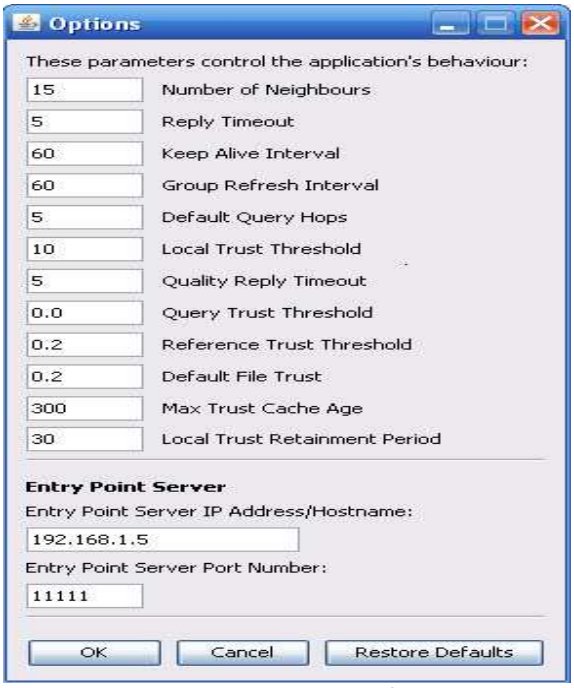

Figure 2. Options interface

The parameters can be roughly grouped into three categories: those that determine the scope of trust management; those that constrain group management and those that affect network behaviour. This facility enables users to adapt the behaviour of the system to prevailing environmental conditions in a network by fine-tuning the types of interaction that takes place between peers.

For example in a relatively stable environment the intervals of time can be increased to reduce unnecessary network traffic. The intervals can also be decreased to ensure that the network is responsive and that the information is up to date. Furthermore, if malicious behaviour is predominant, a high threshold value can be set to exclude malicious peers. The high value can have the additional affect of encouraging tighter group interaction.

3.7 Simulation

A user interface is also provided for the generation and collection of test data through simulation. This involves running the simulation a number of times with a different number of peers, performing a different number of transactions per minute. The system can be tested under different conditions and its behaviour monitored when subjected to different loads. The result of one simulation, for 5 peers where at least 5 actions are performed, is shown in Figure 3. The graph displays the response time for transactions against the elapsed duration of the simulation. It shows that the response time for all the transactions (except one) is within 15 seconds and the average is 2.2 seconds. The variation in the response time is due to the computation of the distributed trust.

The simulation was carried out on a single computer, with multiple processes (5 in the typical test case) for the peers, and a single process for the Entry Point Server. The port on which the Entry Point Server was listening was advertised to the peer processes. The processes were independent, running on the same machine and communicating over TCP/IP.

Once the simulation is started the peers contact the Entry Point Server to request a set of neighbours, which are subsequently notified by the peers of their presence in the network. The simulation controller initiates actions on the peers, at random intervals, causing them to join or leave an interest group, send a message, query for a file, or request a file from the last set of query results. The frequency of the actions is determined by a random sleep period, which is configurable. In the typical test case the time limit was set such that an action would be carried out at least every 12 seconds.

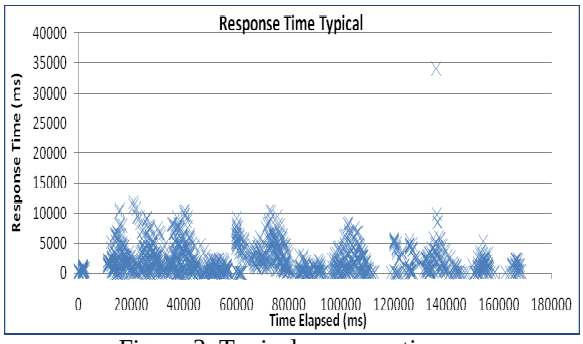

Figure 3. Typical response time

The simulation can support a heuristic process in order to determine the best parameters for use with the trust formula

3.8 Parameter tuning

The tuning process is heuristic, as it requires testing many different parameters under simulated conditions, as described in the previous section. An example of the heuristic process for tuning parameters is demonstrated in Figure 4 for the determination of a suitable value for the Response Time Threshold. This parameter is used to decide whether a response for a transaction is within an acceptable time interval. The results indicate the reliability over time when choosing different values for the response time threshold parameter. The graphs show the effect of four different values on the reliability of a peer:

- 10 seconds: the impression is that the peer is extremely reliable, above 90%.
- 5 seconds: the peer is around 80% reliable.
- 2 seconds: the peer is around 60% reliable.
- 0.5 second: the peer is around 30% reliable.

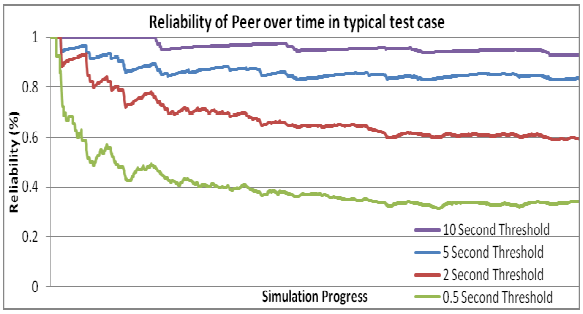

Figure 4. Peer reliability

A value of 5 seconds appears to be the most suitable. In this simulation, it is expected that 80% of transactions will conform to the time constraints. This would lead to the reliability of the peer tending towards 80% of the perceived quality of the files they provide, a relatively fair assessment of their trustworthiness.

The response time can also affect the behaviour of the network. A low threshold for peer selection may lead to a high volume of traffic while a high threshold may restrict the flow of traffic [14].

4 Discussion

In this paper the issue of trustworthy transactions has been addressed by incorporating various competencies into a layered architecture. The system is still under development and a number of issues need to be addressed as part of further work.

4.1 Malicious behaviour

The present approach strikes a balance between full trustworthiness and openness towards new peers. The proposed system does not address explicitly the presence of malicious peers and some features were introduced in order to mitigate their impact:

- The Entry Point Server is used to generate certificates and provide a way of authenticating peers. Certificates assist in dealing fairly with peers and addressing the issue of cold start.
- The trust mechanism, through direct contact (local trust evaluation) and distributed trust (references) will eventually isolate and ostracise the peers that misbehave.
- Most transactions are confined to specific interest groups and are therefore conducted within a relatively trusted environment.

Although it is reasonable to assume that malicious behaviour is rare in P2P systems [15], undetected and unpunished malicious behaviour can however discredit a whole system. The requirement for distributed trust for example can be exploited and misused. Keeping track of peer behaviour over a longer period of time and monitoring changes can help identify harmful patterns of behaviour.

4.2 Trust and reputation

The formula is the result of careful prototyping. It offers flexibility and efficiency. It has the advantage that it requires only four values to be stored for each peer: the number of transactions, the cumulative reliability factor, the cumulative file quality and the number of file transfers. These values can be updated when transactions occur, and the calculation of local trust can be a very fast and efficient procedure. In trust calculation all ratings have equal weight. Further work will be concerned with the assignment of selective weights to different ratings.

In some systems the evaluation of the trust of a peer is divided into two components: the reliability of the peer as a measure of its ability to provide a service of high quality, and its credibility as a measure of its ability to provide trustworthy ratings of other peers [5]. The formula used in this system is much coarser and subsumes credibility within reliability. A two-dimensional approach to peer evaluation is bound to refine trust evaluation in terms of reliability and credibility and will help filter poor recommendations. Although a larger set of criteria would widen and deepen the scope of trust calculation efficiency concerns may constrain the range of parameters that can be considered.

In its present form the trust calculation may also penalise equally all peers, those that are overloaded and those that are genuinely untrustworthy. Work is currently being carried in order to refine the parameters used in the formula. Particular emphasis is put on information gathering that would help discriminate between a peer that is overloaded and one that is incompetent or malicious.

The approach adopted in this scheme conforms to the pull model where information is explicitly requested from other peers and may lead to bursts of activities. A push model, on the other hand, can distribute activities evenly over time and may promote a more proactive approach [14].

4.3 Group management

Membership of a community offers many advantages and transactions within a group are efficient. Both local and distributed trust can be easily calculated. Peers in a community are also likely to interact with

each other much more frequently: it will take less time for the trust threshold to be met and the local trust only is adequate. With respect to quality of service, peers within a community are also likely to share files that are useful to the members of that community and thus will be rated highly. Moreover, this form of socialised trust presents fewer risks [16].
One disadvantage of confining transactions to one community is the potential formation of closed communities. Its members may deny resources to peers who do not belong to a community. One way of addressing this issue is to configure trust parameters to facilitate the interaction with un-trusted peers under specific conditions. This can be achieved by setting the trust threshold of some files very low. The configuration tool can therefore enhance the cohesion of a group by setting a high threshold trust value or create a more open environment for transactions processing between peers.
Another issue concerns the structure of interests. The groups are defined by a flat list of interests. A hierarchical structure may provide more focus and lead to a more refined search for information [17].

5 Conclusion

In this paper a P2P system was presented where group management, trust management and security management are integrated into a layered architecture in order to create an environment favourable to the conduct of trustworthy transactions. An interface for customising group interactions, tuning various parameters and performing simulations was provided as a means of addressing the issues that arise from the dynamic nature of P2P networks. Trust threshold and interval values can be set in response to the prevailing network conditions and to the context in which transactions take place. The formula for trust calculation provides adequate support for useful group interaction and is enhanced by a heuristic process for tuning trust parameters. This combination of group management and parameter tuning is used to mitigate the impact of malicious behaviour.

References

[1] Androutsellis-Theotokis, S., and Spinellis, D. A Survey of Peer-to-Peer Content Distribution Technologies. *ACM Computing Surveys , 36 (4)*, December 2004, pp335-371.
[2] Mekouar L., Iraqi Y. and Boutaba R.: Peer-to-peer's most wanted: Malicious peers, *Computer Networks 50(4)*, 2006, pp545-562.
[3] Cornelli F., Damiani E., De Capitani di Vimercati S., Paraboschi S. and Samarati P., Implementing a Reputation-Aware Gnutella Servent, *Proc. of the International Workshop on Peer-to-Peer Computing*, Pisa, Italy, May 2002, pp321-334.
[4] Resnick P., Zeckhauser R., Friedman E. and Kuwabara K., Reputation Systems, *Communications of the ACM, 43(12)*, 2000, pp45-48.
[5] Wang Y. and Vassileva J., Trust and Reputation in Peer-to-Peer Networks, *Proc. of the 3rd IEEE International Conference on Peer-to-Peer Computing*, Linkoping, Sweden, 2003, pp150-157.
[6] Li, H., & Singhal, MTrust Management in Distributed Systems. *Computer , 40 (2)*, February, 2007, pp45-53.
[7] Khambatti, M., Ryu, K., & Dasgupta, P. Peer-to-Peer Communities: Formation and Discovery. *14th IASTED Conference on Parallel and Distributed Computing Systems*, Cambridge, Massachusett, 2002, pp166-173.
[8] Aberer, K., and Despotovic, Z. Managing Trust in a Peer-2-Peer Information System, *Proc. of the Ninth International Conference on Information and Knowledge Management (CIKM 2001)*, Atlanta, USA, 2001, pp310-317.
[9]Yu B., Singh M.P and Sycara K. Developing trust in large-scale peer-to-peer systems, *Proc. of the First IEEE Symposium on Multi-Agent Security and Survivability*, Philadelphia, USA, 2004, pp1-10.
[10] Wang, Y., and Varadharajan, V. DynamicTrust: The Trust Development in Peer-to-Peer Environments. *IEEE International Conference on Sensor Networks, Ubiquitous and Trustworthy Computing*. Taichung, Taiwan, 2006, pp302- 305.
[11] Condie, T., Kamvar, S. D., and Garcia-Molina, H., Adaptive Peer-to-Peer Topologies. *Fourth International Conference on Peer-to-Peer Computing*, 2004, pp53-62.
[12] Liu, Y., Zhu, G., and Yin, H. A Practical Hybrid Mechanism for Peer Discovery. *International Symposium on Intelligent Signal Processing and Communication Systems*, Xiamen, 2007, pp706-709.
[13] Seigneur J-M., Farrell S., Jensen C.D, Gray E. and Chen Y. End-to-end trust starts with recognition, *in Proc. of the First International Conference on Security in Pervasive Computing*, Germany, 2003, pp251-255.
[14] Ding X., Yu W. and Pan Y., A Dynamic Trust Management Scheme to Mitigate Malware Proliferation in P2P Networks, *Proc. of IEEE International Conference on Communications, ICC 2008*, Beijing, China, May 2008, pp19-23.
[15] Friedman E.J. and Resnik P., The Serial Cost of Cheap Pseudonyms, *Journal of Economics and Management Strategies, 10(2)*, 2000, pp173-199
[16] Pouwelse J.A, Garbacki P., Wang J., Bakker A., Yang J., Iosup A., Epema D. H. J., Reinders M., van Steen M. R, and Sips H.J., TRIBLER: a social-based peer-to-peer system, *Concurrency and Computation: Practice & Experience archive, Vol 20 , Issue 2*, February 2008, pp127-138.
[17] Pogkas, I., Kriakov, V., Chen Z. and Delis A., Adaptive Management of Communities Based on Peer Content and Reputation, *The Third International Multi-Conference on Computing in the Global Information Technology, (ICCGI '08)*, 2008, pp291-296.