A Survey Study on Reputation-Based Trust Management in P2P Networks

Siddharth Maini
Department of Computer Science
Kent State University
Kent, OH 44240
E-mail: smaini@kent.edu

Abstract

In this survey study I have outlined various issues involved in the design of reputation-based peer-to-peer (P2P) system. The survey can be used as a reference guide in a hope to make the P2P systems based on trust management mode more reliable, trustworthy and scalable. The survey presents a study of reputation-based P2P systems currently in use. It takes the example of decentralized unstructured P2P systems such as Gnutella, Kazaa, Fast Track, and SETI etc. The trust management in P2P systems is used in isolating malicious peers and to promote honest transactions between genuine peers. Reputation-based P2P systems have the property to detect such malicious peers using the reputation of the peer(s) providing the resource(s).

Keywords

Peer-to-Peer, Trust, Management, Reputation-based, Networks, Survey, P2P

1. Introduction

A peer-to-peer (P2P) network is a group of Computer nodes which construct their own open unrestricted sharing networks on top of the Internet architecture. Such a system performs application-level routing on top of IP routing. The users (peers) have dual functionality i.e. they are free to join the network and share their resources by functioning as clients when they need to download and they can function as a server when they need to serve resources to other users. Due to the distributed nature of P2P systems there is no central point of attack but such kind of an architecture makes P2P networks very prone to malicious attacks by other peers like sending Trojans, Worms, Viruses, Fake files etc.

Reputation systems provide a way for building trust through social control by utilizing community based feedback about past experiences of peers to help making recommendation and judgment on quality and reliability of the transactions. The challenge of building such a reputation based trust mechanism in a P2P system is to effectively cope up with various malicious behaviors of peers such as providing fake or misleading feedback about other peers. The most general mechanism of establishing trust among peers is using the reputation of the peers providing the resource. The users can rate the reliability of those peers with which they have dealt in the past. A peer requesting

a resource can evaluate the trust ratings of the peer providing the resources using the reliability ratings of those peers which have dealt with the same peer in the past

The main challenge is the way to incorporate various contexts in building trust as they vary in different communities and transactions. Further, the effectiveness of a trust system depends not only on the factors and metrics for building trust, but also on the implementation of the trust model in a P2P system. Most existing reliable reputation mechanisms require a central server for storing and distributing the reputation information. It remains a challenge to build a decentralized P2P trust management system that is efficient, scalable, reliable, and secure in both trust computation and trust data storage and dissemination. Last, there is also a need for experimental evaluation methods of a given trust model in terms of the effectiveness and benefits.

There are many issues involved in the design of the Reputation-based P2P system. This survey presents a comparison of various systems currently in use and the proposed solutions presented by some papers based on reputation management. A typical reputation-based P2P system calculates the trust ratings using the reputations of other peers using different reputation algorithms. Although trust is a value that is associated between two entities, introduction of reputation provides a higher quality of trust evaluation of those peers.

Since anybody is free to join a P2P network there is always a risk of attack by malicious users. So there is a need to isolate malicious peers from other peers. Moreover, authentic peers must be informed about the best downloadable sources in the network. This is done by calculating the trust ratings of a peer which is providing the resource using the reputation of those peers who have already dealt with the peer in the past.

2. Design Issues in Reputation-based P2P Systems

A variety of online community sites have some form of reputation management built in, such as eBay, Amazon.com, SETI project, Morpheus, Kazaa, Slashdot. I have summarized a list of issues involved in the design of reputation-based P2P systems. Four main issues according to me that are important are *General Security Issues; Distributed Systems Security Issues; Social Issues; Performance Issues*

2.1 Security Issues

In general, the present day P2P systems such as Gnutella or Kazaa are not designed to be secure for the users using it. If a user machine running a PC is compromised under a malicious peer attack, it can start giving out false information to a request in forms of returning false routes or false data to a search query. Furthermore, the users have to trust the P2P applications with its code in order for it to operate correctly. Therefore, the nodes must be robust against such malicious attacks. Following is the description of some of the attacks common on P2P systems.

2.1.1 Man-in-the-middle attacks

It is a security threat in which a peer gets between the receiving peer and the sending peer in a P2P network and sniffs the information being sent. It is typically used to be able to read a public-key encrypted conversation. However, these attacks are difficult to carry out.

The attack relies on having complete access to all messages between the two peers wanting to communicate. An example can be two peers A and B who are sharing certain resources. All messages between A and B must pass between the man in the middle M who is logically located between A and B. Upon the start of communication the public keys must be exchanges between A and B. This is where M starts to interfere by creating an own key-pairs for both A and B. These key pairs are distributed back to A and B in a way that M is able to decrypt, read and encrypt messages passing by. A and B will think they are communicating though a secure channel, but only the channel between A and M, and M and B is actually secured and M can read and modify all of their messages.

Gnutella like system is very much prone to such kind of attacks. The most common example is in Gnutella where a Query Hit message is modified by some malicious node in the path. The modified Query Hit directs the downloading request to a non-existent node or an unreliable or a malicious node.

2.1.2 Denial of service (DoS) attacks

The main purposes of the *denial-of-service (DoS)* attacks are to disable or prevent the victim from being able to use its network connection normally. Every peer in a P2P network has to respond to a query from other peers. This requirement to respond can be easily exploited by malicious peers who can collaborate in continually sending matching queries which can eventually make the network connection unreliable or useless. Most of such attacks reply on the weaknesses in the TCP/IP protocol.

A peer is bound to reply to a query message. The system of handling query messages using digital signatures can be easily exploited for Denial-of-Service (DoS) attacks by the attackers who can continually issue high-value queries. Such kind of attacks can also be thought of as group attacks. For example, the P2P systems which make use of digital signatures in order to authenticate the peers, attacker(s) can easily bombard a peer with a high-match queries which will overload the computational system. As a result the system performance will be degraded leading to a high response time.

2.1.3 Buffer Overflows

P2P applications like Kazaa suffer with Buffer overflow vulnerability which can be exploited by the attacker to trigger a denial-of-service (DoS) condition or having his own code to be executed on the attacked machine. Such vulnerabilities make the users prone to many security hazards. In such type of attacks the extra data may contain codes designed to run specific programs or scripts, which can then be used to send information

about the machine to the attacker. This is mostly caused due to poor programming of the P2P applications available.

One example of a buffer overflows vulnerability in FastTrack P2P, which could be exploited to cause a Denial of Service on supernodes and could also compromise them. Supernodes are clients that have a high uptime, large bandwidth, a public IP address, powerful CPU and a large amount of RAM. Supernodes keep tracks of other supernodes and of clients that are logged onto the network. Any user of a P2P client, which is based upon FastTrack, could unknowingly become a supernode. These supernodes can accept incoming requests with information about other supernodes. The packets sent to the supernode may only contain information about 200 other supernodes at the maximum.

If the packet contains information about 203 or more supernodes, it may overflow the allocated buffer. This causes the supernode to crash. It has also been reported, that this could be exploited to execute arbitrary code on the supernode with a 50% success ratio. This vulnerability could be exploited to lay down all P2P networks based upon FastTrack.

2.1.4 Privacy Concerns

While the previous threats require a virus writer to create a malicious program, the simple usage of peer-to-peer connections can prove to be the greatest threat to a corporation. Using peer-to-peer software within a huge environment can create an unforeseen hole in your network security. Such software easily operates within the restrictions of a configured firewall, as the software generally makes outward connections rather than relying on accepting incoming connections.

Users could easily misuse or configure such software to allow outside systems to browse and obtain files from their computers. These files can be anything from confidential data in an email inbox to proprietary design documents. Even if the peer-to-peer network is configured properly, the network should not be used to transfer confidential information. Data is generally passed along the network unencrypted. Such data can easily be obtained by a network-sniffing program. Administrators should consider limiting the usage of peer-to-peer networks due to privacy concerns alone.

Most common example is that of an Adware that is installed automatically when a user is installing the PSP client application on his machine without his/her knowledge. Such Adware programs can be used to track user’s internet usage, his personal information, his IP address etc. The latest trend is a strict attack by Music companies on the users sharing copyrighted information in form of lawsuits.

2.2 Network Issues

The performance of a P2P network is greatly affected by the network to which it is attached to. This is a cause of concern for many daily peer-to-peer users as they are not able to download some resource with ease and speed. There are several factors which can affect the working of a P2P network.

2.2.1 Topological-Changes

If a P2P network is prone to topological changes then it can lead to a consistency problem within the peers & their new changing neighbors as there is no guarantee of fool-proof behavior of the new peers. A topological change might make the overlay network unstable as all the routing information would have to be updated again. All network properties have an impact on how people exchange content on peer-to-peer file sharing networks.

2.2.2 Scalability Problem

Many P2P applications like Gnutella crash if the user is not using high bandwidth. Moreover searching in such systems is still not scalable to a good extent. However, to make use of this self-scaling behavior, a node looking for files must find the peers that have the desired content.

Napster used a centralized search facility based on file lists provided by each peer. Gnutella like P2P systems establish an *unstructured* overlay network of peers. It uses the flooding with random walk approach in order to search. Queries are not sent to a central site, but are instead distributed among the peers. Upon receiving a query, each peer sends a list of all content matching the query to the originating node. Because the load on each node grows linearly with the total number of queries, which in turn grows with system size, this approach is clearly not scalable.

Kazaa like system uses a better *supernode* approach in which the supernodes have higher bandwidth connectivity. Pointers to each peer's data are connected to a supernode so that all the queries are routed to supernodes.

2.2.3 Key Management

Use of public-key infrastructure is prone to the Man-in-the-middle attacks. The public keys can be easily hacked by an attacker and can be used to read the information flowing through the network. One such proposal to reduce such attacks was the use of trusted certification authorities which may or may not be an option in the P2P systems that are totally decentralized. One other solution is to make use of the public key system as pseudonyms. The use of digital signatures would lead to a large amount of computational overhead.

2.3 Social Issues

2.3.1 Treatment of a new Peer

A new peer which joins the reputation based P2P network might be treated differently. An example of one such case sighted in the paper "Reputation-based Trust management for P2P Networks". This paper describes a mechanism where the querying peer groups the responding peers according to the file hash values to identify different file versions from the reply messages. A reputation score is calculated for each group and any random peer from the group is chosen for downloading. The purpose of this was to give a chance to the new peers to build a reputation for themselves.

Also the joining of a new peer might be sighted as a hazard. This new peer might be a malicious peer who changed his pseudonym in order to prevent detection because he behaved maliciously previously. On the other hand this peer can be an authentic peer whose purpose is to actually spread good high-quality resources free of viruses.

2.3.2 Problem of free-riders

Free-riders are peers who use the P2P system only to download resources without making any contribution to the network. These are also referred to as lechers. The result is that they use the network resources for their own mean use due to which other peers have to suffer problems like low bandwidth, frequent disconnection.

Many users in Gnutella-like system are free-riders. So the P2P system should be able to discourage free riding. One example can be of a P2P system which determines the download bandwidth of the peer depending upon the amount of good service it offers to its other peers. One problem is that such a system can also be hacked if somebody is able to manipulate the P2P application installed on his system.

2.4 Performance Issues

2.4.1 Search Propagation and Download Time

One of the main problems in peer-to-Peer systems (P2P) networks is searching and downloading correct information. Due to the decentralized nature of the peer-to-peer systems the searching mechanisms are inefficient. The perfect scenario would be to provide accuracy in information retrieved and discovered objects, and minimum bandwidth production with minimum download time.

In Gnutella2, when a super-peer (or hub) receives a query from a leaf, it forwards it to its relevant leaves and also to its neighboring peers. In this flooding technique these super peers process the query locally and forward it to their relevant leaves. No other nodes are visited with this algorithm. Neighboring hubs regularly exchange local repository tables to filter out unnecessary traffic between them. The number of leaf-nodes per super-peer must be kept high, even after node arrivals/departures. This is the most important condition in order to reduce message forwarding and increase the number of

discovered objects. Also downloading from sources which are near to the peer would prove more fruitful in terms of having a good download speed

In simulation results Gnutella was not able to reduce the amount of bandwidth needed to support many users therefore reducing the scalability. The users (with modem connection) were replied upon to relay information. It does not provide any means to keep the network efficiently knit, so that connections maximize the number of hosts reachable in the fewest hops. Many different searching techniques such as Random Walk with Flooding, Intelligent BFS, and Modified BFS are being proposed as a new solution to make the searching as efficient as possible.

2.4.2 Robustness

P2P systems should have a robust technique to guard against the malicious peers who can collaborate in attacking other peer(s). Also the system should be able to handle flash crowds also called as “hot spots” which is a phenomenon that results from an unpredicted increase in the popularity of an online object. As a result it leads to the performance degradation of a good peer. Presently many P2P systems do not employ protocols to prevent such problems. A possible solution as proposed by the paper “Reputation-based Trust management for P2P Networks” is to actually select a random peer from a group of peers.

3. Detection of Threats

Since peer-to-peer malicious threats still need to reside on the system’s current desktop, a scanning infrastructure can provide protection against infection. However, desktop protection may not prove to be the best method in the future. Should peer-to-peer networking become standard in home and corporate computing infrastructures, network scanning may become more desirable. Such scanning is not trivial since, by design, peer-to-peer transfer of data does not pass through a centralized server, such as an email server.

Systems such as network-based IDS, application-level firewalls as well as gateway/proxy scanning can be used to prevent malicious threats from using peer-to-peer connections that pass inside and outside of organizations. However, peer-to-peer networking models such as Freenet will render networking scanning useless since all data is encrypted. You will not be able to scan data that resides in the DataStore on a system. Detection of threats passed via Freenet type models will only be scanned on the unencrypted file at the desktop just prior to execution. The issue of encryption reinforces the necessity for desktop-based, antivirus scanning.

4. Some Example Proposed Solutions

4.1 Hybrid Centralized Reputation

Person-to-person online auction sites such as eBay, Amazon.com and many business-to-business (B2B) services such as supply-chain-management networks are examples of P2Pcommunities built on top of client-server architecture. In eCommerce settings, P2P communities are often established dynamically with peers that are unrelated and unknown to each other. Peers have to manage the risk involved with the transactions without prior experience and knowledge about each other's reputation.

One way to address this uncertainty problem is to develop strategies for establishing trust and develop systems that can assist peers in assessing the level of trust they should place on an eCommerce transaction. For example, in a buyer-seller market, buyers are vulnerable to risks because of potential incomplete or distorted information provided by sellers. Trust is critical in such electronic markets as it can provide buyers with high expectations of satisfying exchange relationships. Such a reputation scheme can be used in sharing files, making decisions in selecting a peer for a trade transaction in online communities such as eBay & Amazon.com.

4.2 Query-Response Architecture [1]

The paper uses the query response architecture to identify malicious peers & to prevent spreading of malicious content. The protocol proposed is divided into various stages

Resource Query: Query is sent out by a peer to search for a resource. The response message includes the hash of the file being offered.

Trust assessment: The trust ratings for all the responding peers are calculated after grouping them according to the file hashes signifying file-versions. The trust score for each group is calculated using locally stored trust values or using credibility ratings of respondents on a particular peer in case the trust values are not known locally. This is done using a trust evaluation function explaining which is beyond the scope of this survey. Distrust score is also calculated signifying the number of times a peer has behaved maliciously.

File Download: In the end any random peer from the group which satisfies the minimum distrust & maximum trust criteria is selected for download.

4.3 Eigen Trust Algorithm [2]

In Eigen trust each peer i rates another peer j from which it tries to download files by rating each download as either +ve or –ve. Each peer maintains a sum of all his transactions with other peers in a local trust value vector. In order to form a global trust

vector, the local trust values are aggregated around the network and normalized so that malicious peers will not be able to assign arbitrarily high trust values to other malicious peers.

Normalizing a peer's global trust value in this way ensures that all values will lie between 0 and 1. Global reputation of each peer *i* is given by local trust values assigned to peer *i* by other peers. This is weighted by global reputations of assigning peers. These normalized local trust values are aggregated in a distributed environment by asking for opinions about other peers & placing them in a trust vector. In the end the peer having the highest trust value will be selected for download.

4.4 Reputation Computation Agent [3]

The paper uses objective criteria to track each peer's contribution in the system and allows peers to store their reputations locally. Reputation is computed using **DCRC: Debit-Credit reputation Computation or CORC: Credit-Only reputation Computation. DCRC** credits the peer for serving content & debits the peer for downloading resources. On the other hand, CORC *only* credits the peers for serving content but offers no debits. Expiration on score serves as a debit for the user.

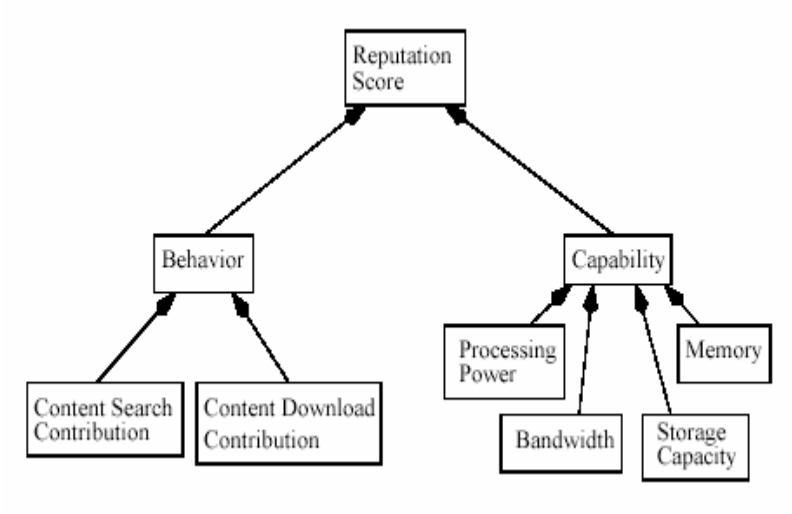
Using Reputation computation agent RCA, reputation can be updated in secure, light-weight & partially distributed manner. This computation maps the peer's behavior & capability pattern. Both schemes track the resources contributed to & by the user by means of a non-negative number of points which represents a *peer's reputation score.*

Both DCRC & CORC offers credits for staying online, query processing & query forwarding. User also have the option of choosing not to track his reputation in which case it will always be visible as 0 to others. Users have their reputations stored locally. So the reputation stored should be stored securely to avoid thwarting attempt. Thus the concept of Reputation Computation Agent was introduced which partially distributed. For good content downloading the type, quality & quantity of the content plays an important role in deciding peer's reputation taking the bandwidth is also taken into account

4.4.1 Main Protocol

- To search the peer generates a query & sends it to all peers it is directly connected to the Gnutella topology
- Peers reply back to this query & also forwards this depending upon TTL
- Peer then selects one of the replies for downloading

4.4.2 Components of the RCA Reputation Score



4.5 Concept of Virtual Currency [4]

In this proposal all nodes are assumed to behave selfishly. Nodes aim to maximize their own reputation in relation to others in the network. It enables a form of virtual currency where reputation of nodes is a measure of their wealth. These features are achieved by developing trusted communities of nodes whose members trust each other & co-op to deal with nodes' selfishness & possible maliciousness.

Such a protocol provides incentives / payoffs to nodes in order to make them co-op. Incentives are provided in form of "Currency" to achieve desired network goals. However, there is an assumption that there would be a centralized entity which would maintain credit/debit values of the nodes. This assumption can be difficult to enforce in P2P. The goal is to maximize their reputation.

Reputation is the key to provide & procure services. As a result the service providers in the on-line economies can be given some compensation for the services/resources they provide. Service providers would be willing to serve nodes with higher reputation to increase there own reputation. The earned reputation makes it easier for service providers to procure services in the future.

4.5.1 Formation of Trust Groups

Another form of proposed solution includes the notion of the nodes forming trusted communities where members trust each other to be good nodes & rely on each other for protection against malicious nodes. If a node x serves node y, then reputation of xTGrp is increased accordingly. If a node provides a good service then the reputation of the provider & all its neighbors in the group in increased otherwise it's decreased for all members.

Every node/peer in a TGrp (Trust Group) shares the same reputation. If the reputation of a TGrp is S with R peers. Then reputation of each peer is R/S. If x & y are members of the same TGrp then xTGrp = yTGrp. Reputation decays with time & is in only incremented if it serves someone outside its group. Thus reputation is updated after ach service transaction.

4.6 Comparison against some Issues

Below is a comparison of the above cited proposed solutions with respect to whether or not they have addressed certain issues involved. These are the most common issues which have been outlined in relation to the P2P networks. There might be many other issues which might not have been addressed by this survey. The sole purpose is to make a distinction between these solutions on the basis of these issues.

	Man-in-the-Middle Attack	**Topological Changes**	**Scalability**	**Vulnerabilities**	**Robustness**	**Treatment of New Peer**	**Free Riders**	**DoS**	**Key Management**
Query Response Architecture	X		X		X	X	X	X	X
Eigen Trust Algorithm			X		X	X	X		X
Reputation Computation Agent					X	X	X		X
Virtual Currency Scheme			X		X	X		X	X
Kazaa						X	X		

5. Example Architectures:

5.1 EBay

EBay has a system to leave a feedback for a person with whom one has completed a transaction previously. In order to make that contribution fair and safe, each member can only affect another member's feedback score by +1, 0, or -1. The feedback score represents the number of members that are satisfied doing business with this member. It is the difference between the number of members who left a positive rating and the number of members who left a negative rating. The feedback score is shown in parentheses next to a member's User ID, for example, eBaytest (3). A rating from a unique member only contributes once to another member's score. If a member leaves three positive ratings for another member (for 3 different transactions), the other member's score increases only by +1.

5.1.1 An Example

A user named **eBaytest** bought and sold a total of 17 items with 9 different members. However, after these 17 transactions, her feedback score increased by only 3. Here is a description of how the eBaytest's feedback ratings were affected after each transaction.

*Members who left a positive***:** eBaytest had only one transaction with each of the following members. They each left her one positive rating (+1). These two positive ratings raised eBaytest feedback score by 2.

*Members who left more than one positive***:** This eBay member had three transactions with eBaytest. All of them were satisfactory, so the eBay member left 3 positive ratings for eBaytest, one for each transaction. This does not mean that eBaytest feedback score increased by 3. Since all three ratings were from the same member, eBaytest feedback score increased only by 1.

Members who had a neutral impact: This member left a neutral rating for a transaction with eBaytest. In this case, eBaytest feedback score neither increased nor decreased but stayed the same.

Members who left a negative: This member was not satisfied with the transaction and so left eBaytest a negative rating (i.e. -1). This decreased eBaytest score by 1.

*Members who left different ratings for different transactions***:** A member can leave only a positive, neutral, or negative rating for another member. Each of these ratings affects the other member's feedback score only once. For example, if a member left two negatives for eBaytest, only one of them will contribute to eBaytest score. However if the same member left two negatives and one positive for eBaytest, the negative rating will count once and so will the positive rating. Subsequent negatives or positives from the same member will not affect

eBaytest feedback score. Take a look at the following scenarios:

A member left eBaytest 1 neutral and 1 positive rating. This affected eBaytest score by (0+1) = +1.

A member left eBaytest 1 negative and 1 positive rating. This affected eBaytest score by (-1+1) = 0.

A member left eBaytest 1 negative and 2 positive ratings. This affected eBaytest score by (-1+2) = +1

A member left eBaytest 2 negatives and 1 positive rating. This affected eBaytest score by (-2+1) = -1

A member left eBaytest 2 negatives and 1 positive rating. This affected eBaytest score by (-2+1) = -1

Final Score: (2+1+0-1+1+0+1-1-1) = **3**

5.2 Amazon.com

Amazon.com allows anybody (already registered with Amazon.com) to leave a feedback on a particular product whether or not the product was actually purchased or not from the same web site. This gives an opportunity for people to do mischievous doings like leaving an incorrect feedback on a product which can affect its sales & marketing.

The person who wants to leave a feedback is asked to rate the product on a scale of 5 Stars and describe the reason behind giving such a feedback. The total feedback is the average of all the feedbacks received on a particular product. Conclusively such a feedback system is not a good example of Trust propagation as one can never know if a feedback is genuine or not.

To deal with this situation Amazon.com provides a system in which a member can report on a particular feedback in case he feels a feedback inappropriate. Amazon.com will then take appropriate action on whether to display the feedback. Also a member can review the feedback history of any other member and can get to have a general idea about him. This might help in deciding whether or not to trust any feedbacks from this member.

5.2.1 Example

Member X gives Product A the feedback rating: 5 Stars
Member Y gives Product A feedback rating: 3 Stars
Member Z gives Product A feedback rating: 1 Stars
Member T gives Product A the feedback rating: 2 Stars
Member P gives Product A the feedback rating: 5 Stars
Member S gives Product A the feedback rating: 5 Stars

Average Feedback Rating for this Product: 1 Star

Below is a screenshot showing the Average Feedback Rating for a particular Product:

0 of 7 people found the following review helpful:

USER, February 20, 2005

Reviewer: **P. Johnson "Tool Nut"** (Pengilly, MN USA) - See all my reviews

REAL NAME

I ordered this P-73 as it is cheaper than the repair estimate on my P-72 owned from 3/03. The P-72 worked fine until it crapped out. Symptoms are a blurred spiked display on the LCD, snaping a pic returns the same crap on the stick. I would have tried another brand (Minolta?) but have sticks and bats on hand for the Sony brand.

Was this review helpful to you? yes no (Report this)

5.3 The SETI Project

SETI (Search for Extraterrestrial Intelligence) is a scientific area whose goal is to detect intelligent life outside Earth. One approach, known as **radio SETI**, uses radio telescopes to listen for narrow-bandwidth radio signals from space. Such signals are not known to occur naturally, so detection would provide evidence of extraterrestrial technology.

Its design consists of millions of clients performing computations on the signal captured by the radio telescopes in Puerto Rico. The servers located at the U.C. Berkeley complex process the results submitted by all the clients. The clients in turn get some credit depending upon their contribution in processing such signals. The client would be listed as a co-discoverer in the event of discovery of an alien signal.

5.4 Decentralized Reputation management in Kazaa

The reputation management function in Kazaa consists of two components namely: *Integrity Rating*, and *Participation levels*.

5.4.1 Integrity Rating

This system allows peers to integrity rate their own files based on whether or not their files have the accurate metadata & are of high quality. It encourages users to delete the corrupted files. There are four levels of Integrity Rating for Files. It is however not obligatory to integrity rate in order to participate in the Kazaa network. When peer integrity rates its files, he will earn double points toward its participation level each time his file is downloaded**.**

Excellent: File has complete Meta data & is of high quality
Average: File has some metadata & is of mediocre quality
Poor: File is of poor technical quality
Delete File: File that should not be shared

5.4.2 Participation Levels on Kazaa

Each peer has a participation Level based upon the quality & the amount of files it shares. It is a number that tells about the way in which the user has uploaded / downloaded files. It can within one of six ranges. It rewards the peer who share many integrity rated files in form of increased bandwidth that they can use to download files form other peers.

The participation level is calculated using the following the formula: **“** $plevel i = uploaded(i) / downloaded(i) * 100$ **”,** where

Plevel(i): participation level of peer i
Uploaded(i): is the amount of data (MB) that peer i has uploaded

Downloaded(i): is the amount of data(MB) that the peer i has downloaded

In the above formula only half the file size will be counted if the uploaded file is not integrity rated.

5.4.3 Issues

There are several issues related to the ratings management in Kazaa. The system rewards peers who demonstrate good behavior, but does not punish the peers who do not or cannot. Malicious peers can also give a high integrity rating to their files even if they are all bogus files. So they keep on generating unlimited number of highly-rated bogus files without being banned from the system.

6. The Future

In truly decentralized P2P environments there are no centralized trusted third parties controlling, storing and providing this information. Instead, the peers provide resources for each other and make trust decisions independently based on incomplete information. The possibility of ephemeral identities and spoofed transactions challenge the reliability of the information available in the P2P system.

Behaving in an expected good manner, a peer can indicate to others its trustworthiness and vice versa. Further, information and evaluations about a peers past behavior, i.e. reputation, plays an important role in assisting other users in their trust decisions. To facilitate these decisions, reputation management mechanisms are being developed to collect and to process the reputation information in peer-to-peer (P2P) environments, e.g. file sharing and electronic market places.

6.1 Balancing privacy and reputation

Identity management is closely related to reputation management. It is possible for peers wishing to protect their privacy to communicate anonymously or by using pseudonyms. At the same time, we should be able to reliably identify the party in question. So, although reputation is important in making the trust decision, it is also a privacy concern when the user related information is stored and the user can be identified.

6.2 Transferring reputation

People form social networks and tend to trust a friend of a friend more than a total stranger. However, building these kinds of trust chains is complicated in a decentralized P2P environment where ephemeral identities are easy to create and no trusted authorities exist verifying the identities. Additionally, good reputation indicating, e.g., experience on evaluating scientific articles is not straightforwardly transferable on good reputation, e.g.,

in selling children clothing. Yet, it seems clear that trust should be at partially transferable between closely related contexts

7. Conclusion

Reputation-based management systems need to address at least the issues stated above in order to make the P2P networks more reliable and robust in the future. P2P networks have already dominated a large part on internet with there popularity increasing day by day at an unmatched extent. What we need to think about is how to make the P2P networks secure, reliable, trust worthy so that its benefits can be fully utilized by many other communities like academia, governments etc. Unless we demonstrate our capability to make the P2P networks fool-proof they can't be trusted. The future of peer-to-peer systems might be seen in the form of Government-to-Consumer (G2C) e-commerce application.

8. Acknowledgements

It would not have been possible for me to finish this survey without the help and mentoring of Professor Dr. Khan. I thank him for pointing me into the correct direction and being very understanding and patient.

9. References

[1] "Reputation-BasedTrustManagementforP2PNetworks" by Aydın Selcuk Ersin, Uzun Mark, Resat Pariente
[2] "The Eigen trust Algorithm for Reputation Management in P2P Networks" by Sepandar D. Kamvar, Mario T. Schlosser, Hector Garcia-Molina , May 2003, Proceedings of the twelfth international conference on World Wide Web
[3] "A Reputation System for Peer-to-Peer Networks" by Minaxi Gupta, Paul Judge, Mostafa Ammar , June 2003, Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video
[4] "Reputation Management Framework and its use as Currency in Large-Scale Peer-to-Peer Networks" by Rohit Gupta, Arun K. Somani, August 2004, Fourth International Conference on Peer-to-Peer Computing (P2P'04)
[5] "A Matter of Trust: Reputation Management in Peer-to-Peer Networks" by Joseph O. Patterson
[6] http://citeseer.ist.psu.edu/stavrou02lightweight.html
[7] http://www.cs.umd.edu/class/spring2001/cmsc433-0201/Projects/p5/p5.html
[8] "A Survey of Trust in Internet Applications" by Tyrone Grandison and Morris Sloman, IEEE Communications Survey
[9] "Content Availability, Pollution and poisoning in File Sharing Peer-to-Peer Networks" by Nicholas Christin, Andreas Weigend and John Chuang

10. Scope of Survey

The scope of this survey is limited to the papers selected on the basis of interest from the ACM Digital Library or IEEE Library. As the research on peer-to-peer networks are being done on wide basis, it is impossible to cite the work of each and every researcher in this field of study.