SECURING A CLOUD-BASED COMMUNITY TRUST STORE FOR LOCAL P2P E-COMMERCE

A.H. Fauzi and Hamish Taylor
School of Mathematical and Computer Sciences,
Heriot-Watt University,
Edinburgh, United Kingdom,
E-mail: {ahf4, h.taylor}@hw.ac.uk

ABSTRACT

Peer-to-peer(P2P) trading applications have much lower costs than conventional client-server systems because they avoid the use of third party service providers. However, a P2P trading application has to be secure and equitable to the peers involved in transactions. This cannot be achieved without the use of some kind of reliable security service provider of trust data. We discuss using a community trust store to host trust related information to support a community based P2P local trading application. It is managed collectively by the peers and is hosted in cloud storage. Community trust stores provide trust data and community assurance to the peers. The trust store only supports mutual authentication, community voting and reputation services. The P2P messaging systems supports all other forms of communication and interaction.

KEYWORDS

P2P, security, generic requirement, serverless, e-trading

1 INTRODUCTION

Several ways of trading over the Internet have evolved over time. One pattern involves a party who sets up a market place, specifies a trading model and provides supporting services. Other parties then trade within that context by those rules. This pattern is typically implemented by the organising party setting up an e-commerce system on the client-server paradigm and running servers to support trading on its model. Other parties then employ a client to interact with these servers and financially compensate the service provider for providing the service. Successful examples of this pattern include eBay, Amazon and Alibaba.com.

A different pattern involves a group of parties who get together to establish a common trading framework and share responsibility for providing supporting services. On this approach no single party or small minority of persons organise the market place. Organisation is decentralized and communally achieved. The trading model is supported by common use of the same software in a deployment configured to serve communal interests. This paper adopts this latter peer-to-peer approach and specifies a peer-to-peer design framework to achieve it.

Peer-to-peer technology is generally cheaper to run and can be more scalable compared to conventional client-server systems. However, the participation of peers in a peer-to-peer network is often unreliable, as each peer can be online at different times and delays can be expected with real time peer-to-peer transactions.

All e-commerce involves risk taking by participating parties. These risks are various but include being cheated, being taken unfair advantage of and being deprived of fair opportunities to trade advantageously. E-commerce traders need assurance that they can mitigate these risks to a sufficient degree otherwise it becomes unwise of them to engage in such e-commerce. Part of that assurance is provided in P2P e-commerce systems by their trading model. Another part is provided by its security services.

The latter includes trustworthy means of identifying trading parties and reliable storage of all trust data needed during trading to evaluate transactions and judge the trustworthiness of trading parties. In our approach a Community Trust Store(CTS) stores all trust related data of the trading parties, including peer identity credentials, trading contracts, trading outcomes and reputation reports. Due to the uncertain availability of the personal peer computing platforms, the CTS need to be hosted by a means which is available whenever it is needed.

A cloud is one solution to this problem. It is able to host a CTS and support continuously accessible storage of trust data for trading activities for P2P e-commerce trading parties. A cloud provides a cheap solution for hosting a CTS compared to one or more conventional dedicated servers provided by a trusted third party e-commerce service provider. Several popular cloud services offer to host applications with a small data footprint, modest throughput and moderate use of bandwidth for free. The CTS is of this kind.

In this paper we will confine our attention to the overall design and security issues for P2P e-commerce to support local trading for low valued goods. It is less problematic than P2P e-commerce in general because the items which are being traded are low valued and the proximity of the trading parties means that the buyer can inspect the item before buying and the parties can exchange the money and goods directly and at the same time. So it avoids the problems of services like eBay of buying without inspecting at first hand, unsynchronised exchanges of money and goods, insecure remote payment, high charges for remote delivery of goods and the risks of suffering a large loss in a single transaction through fraud or mishap. It can also support trading for local services.

2 SCENARIO OF TRADING IN P2P-CTS APPLICATIONS MODEL

A trading scenario illustrates how P2P local trading applications can exploit a CTS. Generally, the trading process takes several steps:

1. Buyer (Bob) searches for items and seller(Sue) advertises a desired item through the P2P messaging service of the online trading forum. After looking around they find each other.

2. Bob and Sue look up reputation reports in the CTS stored in the cloud on each other's trading history.

3. Via the P2P messaging service Sue and Bob agree on a price and to trade money for the item at a meeting subject to a satisfactory inspection. They draw up a contract with these terms.

4. Bob signs the contract, authenticates with the CTS and submits the trading contract. Sue authenticates with the CTS and co-signs the contract submitted by Bob.

5. Bob and Sue meet for inspection of the item and after inspection decided to go ahead with the trade on the agreed terms.

6. Bob and Sue authenticate in turn with the CTS and co sign a trading outcome report that they lodge with the CTS.

7. Bob authenticates with the CTS, is authorized to access the trading contract, and adds his reputation report on the outcome.

8. Sue authenticates with the CTS, is authorized to access the trading contract, and adds her reputation report on the outcome.

   Alternative outcomes for step 5 could be that no trade is done or the trade goes ahead with an adjusted price or other altered terms. Contract adjustments would have to be jointly signed and lodged. Whether the trade goes ahead or not, Bob and Sue still need to provide reputation reports on each other's behavior. Reputation reports stored with the CTS become a point of reference for others that might wish to trade with Bob and Sue in the future.

3 SECURITY ISSUES FOR P2P-CTS

There are several factors which create vulnerabilities to threats from attackers for P2P-CTS systems. As the CTS is stored in the cloud and accessed by the Internet, attackers can try to access the CTS and its content from anywhere. Furthermore, there is

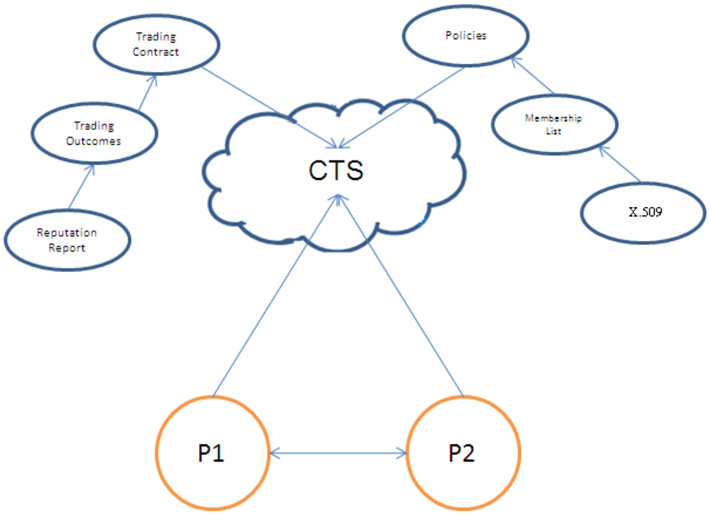

Figure 1: Overall View of the P2P-CTS System

no central authority that controls the CTS. A self-managed community system creates opportunities for attackers to blend in with the community and orchestrate actions with intention of defrauding other peers. Decision making and rules that are made collectively by the P2P community can be subverted if malicious peers can gerrymander sufficient votes to alter membership or pervert rules to subvert fair and open trading.

3.1 What Needs to be Secured and Protected?

In the P2P-CTS, an important resource is reputation reports on prior trades which are stored in the cloud. It is valuable background for peers before engaging in any trading transaction with another peer. Looking up that peer's identity on the CTS will give peer access to all recorded trades outcomes and counterparty evaluations on these trades.

The trust data items that need to be secured and protected by the CTS include:

- Reputation reports
- Identity of peers
- Trading contracts and outcomes
- Membership status
- Community rules

Trust data stored in the cloud is co-related with other data but cannot be trusted piecemeal. It has to be evaluated and cross-checked with other data before deciding to trust it as a whole.

3.2 Threat Model for P2P-CTS

We anticipate the following threats to the CTS below based on the STRIDE security model:

- Spoofing identity - Unauthorized use of another peer's identity to access the CTS. Any malicious act by the attacker will be blamed on the peer whose identity is stolen. Identity spoofing can happen when security secrets such as X.509 private keys are compromised.
- Identity churn - To escape their poor reputations peers may acquire a new identity with no prior trading history and join as a fresh member with a novice's presumed good trading intentions.
- Unauthorized tampering with trust data - Trading contracts and outcomes stored on the CTS should only be able to be added, modified or deleted by the trading parties involved with their joint agreement. Any modification by

anyone other than the parties involved is unauthorized modification of data in the CTS. Modification of such data by either party without the other‘s consent is also unauthorized modification.

- False repudiation - Denial of the act of accessing or updating trust data without the CTS being able to prove the user did so can happen with a poorly designed or insecure logging system.

- Unauthorized information disclosure - Access to data in the CTS by unauthorized parties such as non-members. It can also involve breach of privacy for peers who agreed to confidentiality about the trade.

- Denial of services - Although the CTS are not meant for frequent access, it has to be available when needed. A DOS attack on the CTS could cause its unavailability to peers. Some cloud providers bill cloud access based on connectivity by any party to that bit of the cloud so a DOS attack on the CTS might cost more than just service unavailability but also large charges from the cloud provider to the peer community.

- Unauthorized elevation of privileges - It could happen if peers such as the CTS founding members gain privileged access to the CTS and are able to modify any data inside the CTS. Poor design of the bootstrapping process, authentication, authorization and access control would contribute towards this threat.

- Fraudulent Collusion - Two or more peers may collaborate to fake trades, outcomes and reputation reports to whitewash their trading histories as a prelude to cheating others.

- Sybil Attack - When a peer uses multiple identities (existing identity, alternate identities and third party identities) to manipulate or modify trust data or generate bogus trust data.

- Man in Middle Attack - Malicious peers intercept messages between two communicating peers, modify messages and masquerade as peers to others.

- Blackening reputations - By issuing false reputation reports or unauthorized modifications of existing reputation reports to ruin the reputation of a peer in good standing.

- Whitewashing reputation - By deleting or modifying existing bad reputation reports and replacing them with a good reputation report.

- Off Record Dealing - Peers may try to avoid acquiring a poor reputation in dodgy deals by suggesting to counterparties that recording their transactions is not worth the effort.

- CTS Subversion by Cloud Hosting Service - technical support staff of the cloud service may interfere with the operation, software or data stored in the CTS.

The possible threats might also arise from a combination of two or more of the above threats which could make it more complex to handle.

3.3 Security Requirements for P2P-CTS

Security requirements for the P2P local trading system can be distinguished into the following aspects [1] :

- Access control - only members are allowed to participate in the trading forum community. Non-members are kept out from P2P-CTS community facilities.

- Authorization - membership is controlled and limited by community collective decision making.

- Integrity - trust data consistency in the CTS is preserved and protected.

- Confidentiality - trust information is protected against disclosure to unauthorized peers.

- Non-repudiation - the originator of a message, trading contract or reputation report cannot credibly deny its role in its origins.

- Availability - members have access to the CTS whenever they need it.

3.4 Security Solutions to Resolve Security Issues in P2P-CTS

Proposed solutions to satisfy the security requirements of the P2P-CTS are as follows;

- A mechanism is provided to securely bind a peer's real identity with their trading identity and provide assurance of it through a digital signature. In order to record transactions during trading, outcomes of the trading, reputation reports and membership status, the peers have to bind their personal identity with the trading identity used in the P2P-CTS system. The trading identity is initially established using an X.509 certificate signed by the CTS and its corresponding signature. The identity of peers can also be re-verified in the same way during a meet up between a buyer and a seller.

- Protecting the trust data of the system using identity credentials by limiting access only to authorized members of the P2P-CTS. Without the trust data, the P2P-CTS will become useless in terms of securing transactions against fraud. As each trading transaction is required to be recorded, the recording helps prevent malicious activities from existing members who want to hide their dodgy dealing from affecting their current membership reputation.

- Securing communication among peers and the cloud using encryption. The data or messages shared, transferred and sent among these parties should be digitally signed and may also use encrypted communication channel (SSL/TLS) to pass messages to ensure their integrity and confidentiality. X.509 certificates and private keys can support these requirements.

- Transactional data in the store will be jointly signed before being stored in the CTS. It will make sure that the data has integrity and is undeniable by peers that sign the trading contract or outcome report. Reputation reports by one party on another party's trading behaviors will be individually signed. In peer community membership, community decision making decides the type of rules and their parameters to control membership creation, withdrawal and renewal and the software enforces these constraints.

- Badly behaving P2P members will be reported to the community. Cancellation and revocation of their membership could be done immediately by inserting the peers identity in the CTS certificate revocation list after an ad hoc community vote. More usually it will be done by a community refusal to renew their membership when it falls due. Lack of sufficient sponsors or a sufficient weight of blackballers will ensure this. The required thresholds of each will be a community rule making matter.

- Enforcing the security of the cloud itself with a strict logging and membership system and strategies of backup and replications. The cloud that is being used to store the information has to be secure against threats and able to be accessed without disruption by peer members. Services can be unavailable because of denial of service attacks directed at the peers and the CTS. Their unavailability will discourage people from using a P2P-CTS trading system. Judicious choice of a reputable hosting cloud service can provide reasonable protection against malicious manipulation of the CTS by their technical personnel.

3.5 X.509 Certificate for P2P-CTS

A self-signed X.509 certificate or Certificate Signing Request (CSR) created by a peer applying for membership will be signed by the CTS's private key once his sponsors have satisfied community membership proposal rules and attest to having carried out due diligence on the X.509 certificate.

The certificate and private key can be readily created using Java "keytool" and "openssl" software. The X.509 certificate allows secure connections to be established between the peers and the cloud store

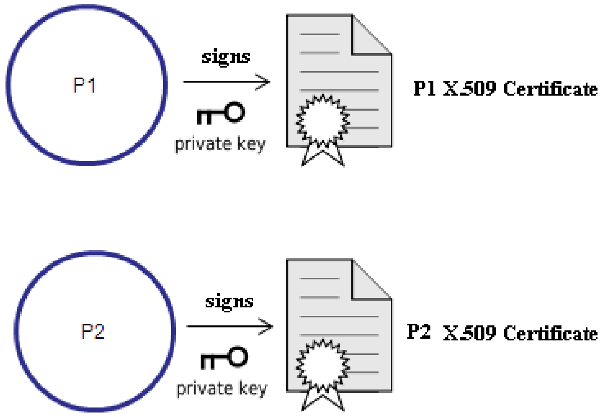

Figure 2: Certificate X.509 P2P-CTS System

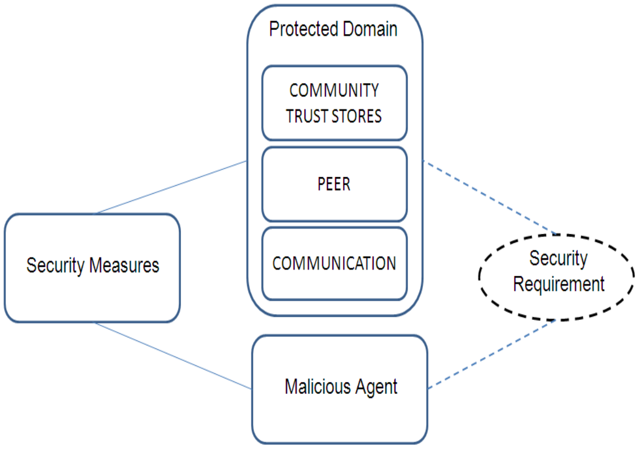

Figure 3: Security Requirement P2P-CTS System

using SSL/TLS with two ended authentication. It can also be used to digitally sign messages, trading contracts or reputation reports.

In the public key infrastructure system, the digital signature used by the CTS to sign the peer's certificate attests that the peer's X.509 certificate is valid and contains correct information.

In the P2P-CTS it will also attest their current membership status.

4 IMPLEMENTATION MODEL FOR P2P LOCAL TRADING AND COMMUNITY TRUST STORES

Peer-to-peer computing benefits local trading applications since it is cheaper to participate in compared to conventional client-server e-commerce applications. No third party service providers needs to be compensated for providing support services. The overall proposed framework for using P2P and cloud computing technologies in e-commerce applications is depicted in Figure 4. P2P is used as the whole network infrastructure and cloud computing as a subnetwork infrastructure for supporting the CTS. Details of this framework are discussed further a the previous paper [3]. The CTS enables peers to:

- jointly record trading contracts with the CTS
- jointly add trading outcomes based on agreed contract
- record reputation reports on trades individually
- store data on the status of peers such as membership of the P2P-CTS system

In local trading, important issues are how well traders honour contracts between trading parties and how well contracts protect the interests of both parties fairly and securely. In direct selling of pre-owned goods, a peer would typically expect to inspect the item before going ahead with buying it. However for never previously used services offered by a vendor, peers cannot assess the quality of a service before it is rendered and have to rely on feedback or testimonials from other peers that have used that kind of service from that vendor before. For example, in relation to a local cleaning service offered by a trader, other peers would expect to be able to consider feedback from peers that have used that trader's service before agreeing to hire that cleaner. Different types of trade will have their own distinct requirements and challenges.

Cloud computing is a cheap and scalable approach to support applications like a CTS. A cloud service such as Google Cloud Storage(GCS) stores and accesses data on Google's infrastructure combining the performance and scalability of Google's cloud. Google Apps Engine(GAE) is a web application service provided by Google which allows the

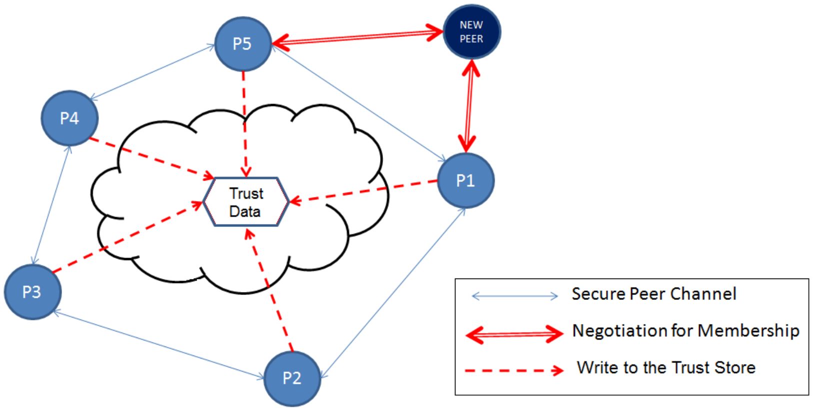

Figure 4: The Framework for P2P e-Commerce and Community Trust Stores

building and hosting of application on platforms provided by Google infrastructure [4]. Using applications built on the cloud provides robust and usable ways for peers to access the cloud.

Figure 5 shows the combination of both systems.

Peers have public keys [5] and CTS signed X.509 certificates to identify themselves. Peers sign their messages with their X.509 private key and certificate as well as use them to provide secure two end authentication SSL/TLS connections with the CTS.

5 BOOTSTRAPPING P2P e-COMMERCE COMMUNITY APPLICATIONS

A P2P community is a group of peers with a common purpose, similarity, and interest [6]. In a P2P e-commerce community of the kind we envisage, the group of peers collaboratively communicate with each other and manage their trust store to achieve the common purpose and interest of trading with each other. They have a set of policies and rules agreed by the community to ensure trading is done fairly and reasonably.

The community starts with two or more peers creating a storage space in the cloud and loading the software application to manage it. Policies and rules are outlined, updated and enhanced with agreement from the two or more members which are needed to authorize them and stored in the CTS. Once the CTS is set up in the original user‘s name, the software then changes his credentials to random values so that no user has privileged access to CTS operations. From then on all peers who are members are equal and changes to store data can only be made in accordance with what the software permits. Then the community can grow by inviting more peers to join the community membership with the recommendation or at the invitation of existing members. Membership must be agreed in accordance with existing community policies and rules. Membership can be revoked if an existing member breaches the policies and rules, and is voted out by a certain number of peers as outlined in the community rules.

In order to join the community, a new member has to agree to abide by existing policies and rules of the community. However, once it has joined the new member can gather support among other peers and use it to abandon or modify existing rules, create new rules or even vote for other peers to be cast out. This problem is unavoidable in such a type of self managed system. A time honoured solution is to designate key rules as constitutional and make changing them possible only with hard to obtain widespread assent among the community.

Peer logs not only provide evidence of good behavior and good reputation but also evidence of malicious activities such as concerted fraud, whitewashing [7] and reputation blacking.

Free riders [8],[9],[10] are another problem for P2P systems. In the P2P-CTS application, free riders would include peers who use existing data like contract outcomes and reputation reports in the CTS but are reluctant to contribute anything back to the

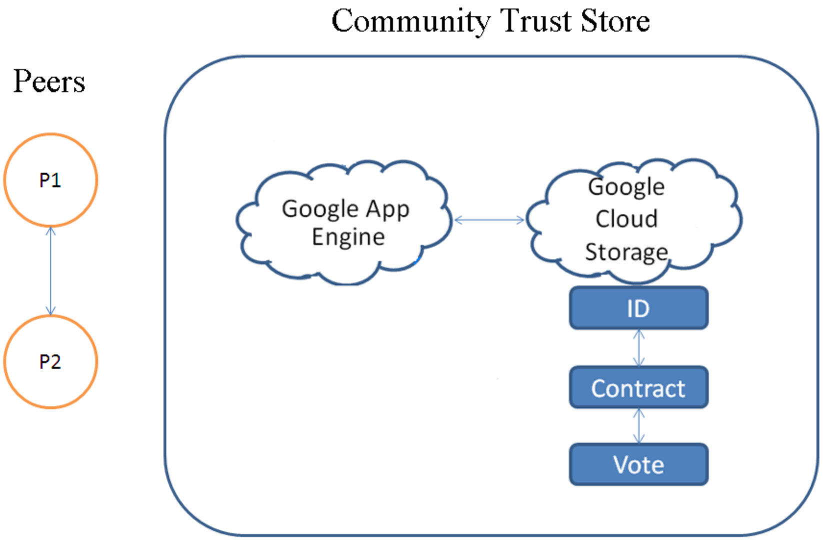

Figure 5: Overview of P2P and Community Trust Stores

CTS such as writing reputation reports or transaction outcomes after trading with other peers. We propose using positive and negative incentives as ways to overcome the free rider‘s problem. Members who fail to contribute will have their membership expired eventually for inactivity. A rule such as membership without contribution to the shared data or reputation module in a given period like three months can be expired automatically. Furthermore, the trading forum may also choose to have a policy such as to forbid private deals between members without using the P2P-CTS. It could be supported by an anonymous whistle blower reporting facility.

5.1 Membership Management

The preferred P2P-CTS membership policy is a closed membership scheme with new peer membership being based on the invitation and voting of existing members. The membership shall be limited to a period such as a year and then be open to being renewed by lightweight or full procedures. Inactive membership can be expired after a shorter period such as three months. Community voting support includes:

1. creating a new member
2. removing an existing member
3. changing community rules
4. renewal of membership

A lightweight process should exist for renewal subject to the member being in good standing (i.e. low percentage of complaints among recorded trades). Those lacking a good reputation might be required by community policy to need a larger amount of voting members to renew their membership. Voting rules for the P2P-CTS should specify the quorum required for a successful application, whether a number of sponsors are also required and whether blackballing is allowed and by what threshold.

5.2 Membership Expiration and Incentives

The membership expiration rules are important in this application to encourage active contribution from peers that use the system. The disadvantages of having too many inactive member includes consumption of physical space in the CTS, management burden of membership and effects on voting percentages. Membership needs to be expired after a set period to weed out non-participants. Apart from membership expiration, in order to encourage peers to contribute and participate, incentives should be introduced for members to trade with each other. P2P systems generally use a variety of incentives including financial rewards [11] and social recognition such as star ratings of members. However, P2P-CTS is different from other P2P systems such as BitTorrent. BitTorrent file sharing space encourages users to contribute by rewarding the activity [12]. The more a peer shares its space, the more it can download from others [13]. As the space usage of the P2P-CTS is not the issue, other incentives are recommended instead. By giving incentives, it should be possible to encourage peers to behave well in the community and deter them from doing bad things [14]. One viable type of incentive could be limited protection against membership exclusion or extra voting power (such as one vote that counts as two) for members who have a recent trading history with good average reputation report ratings. We do not expect all P2P local trading communities will want to operate the same rules or incentives, so our implementation is designed to support a policy module which can allow each community to set its own rules within a number of supported ranges of variation.

5.3 Scheme for Handling Policies and Voting in P2P-CTS

In order to vary existing policies, a peer has to propose doing so to the community for approval and consent. One method for getting approval is by a voting process. Such a process was proposed by the article collaboration project in PeerVote [15].

Although the aims of the voting process are the same, the approach for P2P-CTS will differ due to its different requirements and attributes.

P2P environments create a challenge to support the casting of votes at the same time but this voting scheme has to be able to support this decentralized environment.

The CTS will act as the mediator in the voting process along the following lines :

1. Peer applies to CTS to act as mediator in a ballot and supports application with required number of sponsor signatures, policy variation details and summary supporting argument.

2. CTS checks admissibility and announces a vote and its policy based term such as its duration positive threshold minimum and negative threshold maximum via P2P messaging system.

3. Peers discuss issue among themselves during that period. Votes are limited to only two choices, agree or disagree. Non-voting members are considered to be neutral.

4. Peers vote by giving their choice(agree or not) and signing it with their digital signature to CTS

5. The voting can end early if positive threshold is reached or negative threshold maximum are reached.

6. When the voting duration ends the CTS adjudicates the result, announces it and implements it. Implementation means changing a policy parameter, membership status or mechanism parameter.

5.4 Reputation Report Scheme

In a P2P e-commerce community, reputation is important as it help other peers to judge the reliability and trustworthiness of a peer. If peers are highly reputable, the peer can get good feedback from others and positive testimonials from them. The reputation report scheme includes reputation reports on previous trades, testimonials and complaints. Other peers will refer to reputation reports on previous trading deals before making any trading engagement with a peers. Apart from the reputation report contents, it also has other factors that might affecting reports such as the most recent reports and numbers of previous transactions involved.

Reputation reports on individual deals cover several factors such as the satisfaction level of participating peers, perceived trustworthiness of peers, terms fulfilled as agreed in previous agreement and additional or extra bonus which might be included in the deals.

This reputation report would be expected to use a multi dimension rating scheme which gives marks on a finite scale between 0-10 and free text comments. In order to encourage peers to have good reputation community star recognition or VIP members can be added to distinguish members who have passed certain limits of positive overall feedback. The reputation report can be sorted in term of total average marks, positive rates, most recent transaction and number of star recognitions. The history of activities related to reputation reports such as numbers of peers being rated or numbers of peers' reputation report made by others. Rating activities shows the activeness and participations of ones peers involving writing reputation reports to others after meeting of item inspections and writing the transaction outcomes.

Summary of the overall reputation can show the trustworthiness of a peer. Two problems with reputation systems are whitewashing, where peers use illegitimate ways to improve their reputation such as by suppressing bad reputation reports on their trades and badmouthing, where peers try to ruin good reputations of other peers by creating bogus bad reputation report on other peers. Both are a challenge that need consideration in the design of the P2P-CTS including ways to protect the reputation report integrity by hardening the modification process and enforcing access control to the reputation reports.

6 RELATED WORKS

Several advanced approaches have been suggested regarding how to evaluate trust data concerning the reputation of peers [16], [17], [18], [19], [20], [21]. These approaches have several weaknesses and challenges. However, most of the proposed reputation systems have over by limited dimensions which are unable to evaluate and determine the peers' reputation informatively. Too many dimensions also create another problem of too much data to key-in and also making reputation assessment too complicated for practical use in a P2P trading system.

Existing reputation report systems also largely assume that peer feedback is honest and non-biased. This is not always true because there is always the possibility that although a transaction went well between peers, a peer still can get a low rating from each others which doesn't reflect the transactions. There is also possibility that a good peer maybe unfairly rated badly by another peer to ruin the peer's reputation. It is an issue of the credibility of the feedback given by the peers. For example, how do we deal with trusting reputation reports contributed by a peer who previously has had dodgy trading transactions and bad reputation reports. Qualifying or annotating reputation reports by overall ratings of the report giver's own reputation is one way forward.

Using a cloud as storage to store trust data also creates challenges of its own. There are cloud based products such as Dropbox [22] and Google Cloud [23]. There are also P2P storage alternatives such as Wuala [24], PeerStore [25] and OceanStore [26]. For our system, selection of the cloud provider is based on their good reputation, wide accessibility and minimal subscription cost.

In our approach, there is no single controller or super user of the cloud account nor are there a subset of privileged users. The cloud is publicly available and accessible to community members which is the main reason for using a cloud to allow accessibility of the trust data stored in them. By using existing cloud products peers do not need to worry on how to setup the cloud. The main concerns are related to how control of the information stored in the cloud is determined by the provider.

7 DISCUSSION AND FUTURE WORK

Several key aspects distinguish a P2P-CTS system from classic client-server e-commerce applications. They include;

- Trading policies are controlled by peers and enforced in the main by CTS mechanisms
- Collective decision making by community members
- Joint signing of contracts and trading outcomes
- Dynamicity, evolutionary and expandability of overall operations

- Using cloud service as the trust data storage platform

The P2P-CTS applications framework is being experimentally implemented on a third party platform so that it can be purely managed by the peers. The implementation success depends on the peers and whether they endorse the functionality and strength of the security mechanisms put in place. We intend to calculate and measure that assurance using methods like voting.

The same goes with trust in the use of P2P-CTS stored reputation reports or recommendations by peers. The number of votes and ratings awarded by the peers can be aggregated to assess community opinion on the CTS-P2P system.

Transparent logging in the CTS enables evidence to be followed and traced until the source of any issues is revealed. Peers can independently check the evidence and make their own decisions based on their findings.

Reporting on the outcomes of transactions and reputation reports are one of the key factors for the implementation. Issues such as misleading for sale advertisements, improper categorization of ratings, misinterpretation of comments, accidental and unintentional grading are human factors that might affect P2P-CTS application outcomes apart from the protection mechanism of the CTS. They can be mitigated as part of further enhancement for the P2P-CTS application.

The P2P-CTS design includes support for a convenient Public Key Infrastructure(PKI) and X.509 certificate generation. The CTS will have its own private keys and self-signed X.509 certificate. All transactions with the CTS will take place via SSL/TLS with two ended authentication by using X.509 certificates.

In the near future, a demonstrator of a P2P-CTS system will be validated with a series of test to prove its viability to support local P2P trading.

8 CONCLUSION

From the list of threats which would be encountered by a CTS, we have presented suitable security strategies able to protect the contents of the community trust store and provide reliable access to it. Apart from protecting the contents in the CTS, we have outlined the methods being employed to protect the identity of the CTS itself. We have also described peer access to the CTS service including handling a denial-of-service attack. Although the P2P-CTS is hosted by a third party platform in the cloud, it is still managed and monitored by the participating peers' community. Monitoring changes in the P2P-CTS by the peers for unusual patterns will be a key part of collaborative efforts towards ensuring the security of the CTS.

Community assurance by the community of peers underpin use of the CTS-P2P. Collective and collaborative decisions and actions to overcome problems and administer issues related to the CTS will ensure increase trust and confidence towards use of the application.

However, all the trust data stored in the cloud is co-related with each other and cannot be trusted piecemeal. It has to be contextually evaluated and checked before deciding to trust it.

In conclusion, we have presented security requirements and an overall design for a Cloud based Community Trust Stores for local P2P e-commerce. The requirements allow the community to vary the trading rules within set bounds to adapt the CTS to the local community needs and preferences. The CTS software will enforce these requirements and determine the scope of variation of trading rules.

References

[1] ITU-T. ”Recommendation X.800: Security Architecture for Open Systems for CCITT Applications”, (1991).

[2] Houser, D., and Wooders, J. : ”Reputation in Auctions: Theory, and Evidence from eBay”. Journal of Economics & Management Strategy, vol. 15, pp. 353–369. Blackwell Publishing (2006).

[3] Fauzi, A.H., and Taylor, H. : ”Community Trust Stores for Peer-to-Peer e-Commerce Applications”. In Proceedings of Informatics En-

gineering and Information Science, vol 251, pp. 428-442. Springer, Heidelberg (2011).

[4] Fisher, P., Pant, R., and Edberg, J. : ”Cloud Computing: Assessing Azure, Amazon EC2, Google App Engine and Hadoop for IT Decision Making and Developer Career Growth”. Apress (2010).

[5] Zimmermann, P.R.: ”The Official PGP User’s Guide”. MIT Press, Cambridge, MA, USA (1995).

[6] Vassileva, J. : ”Motivating Participation in Peer to Peer Communities”. Engineering Societies in the Agents World III. Lecture Notes in Computer Science, vol. 2577, pp. 18-23. Springer Berlin, Heidelberg (2003).

[7] Kudtarkar, A.M., and Umamaheswari, S. : ”Avoiding Whitewashing in P2P Networks”. In Proceedings of the First international conference on COMmunication Systems And NETworks, pp. 115-118. IEEE Press, NJ, USA (2009).

[8] Feldman, M., Papadimitriou, C., Chuang, J., and Stoica, I.: ”Free-riding and Whitewashing in Peer-to-peer Systems”. IEEE Journal on Selected Areas in Communications, vol. 24, no. 5, pp. 1010-1019 (2006).

[9] Feldman, M., and Chuang, J.: ”Overcoming Free-Riding Behavior in Peer-to-Peer Systems”. ACM SIGecom Exchanges, vol. 5, no. 4, pp. 41-50 (2005).

[10] Karakaya, M., Korpeoglu, I., and Ulusoy, O. : ”Free Riding in Peer-To-Peer Networks”, IEEE Internet Computing, vol.13, no.2, pp. 92-98 (2009).

[11] Golle, P., Leyton-Brown, K., Mironov, I., and Lillibridge, M. : ”Incentives for Sharing in Peer-to-Peer Networks”. Electronic Commerce. Lecture Notes in Computer Science. vol. 2232 pp. 75-87. Springer Berlin, Heidelberg (2001).

[12] Cohen, B. : ”Incentives Build Robustness in BitTorrent”. In Proceedings of the 1st Workshop on Economics of Peer-to-Peer Systems”, pp. 978-982, Berkeley, CA, USA (2003).

[13] Anagnostakis, K.G., and Greenwald, M.B. : ”Exchange-based Incentive Mechanisms for Peer-to-peer File Sharing”, In Proceedings of the 24th International Conference on Distributed Computing Systems, pp. 524-533 (2004).

[14] Zghaibeh, M., and Anagnostakis, K.G. : ”On the Impact of P2P Incentive Mechanisms on User Behavior”. In Proceedings of NetEcon+IBC, ACM, pp.1-23, San Diego, California, USA (2007).

[15] Bocek, T., Peric, D., Hecht, F.V., Hausheer, D., and Stiller, B. : ”PeerVote: A Decentralized Voting Mechanism for P2P Collaboration Systems”, AIMS 2009, pp.56-69 (2009).

[16] Teacy, W. T. L., Patel, J., Jennings, N. R., and Luck, M. : ”Travos: Trust and Reputation in the Context of Inaccurate Information Sources”, Autonomous Agents and Multi-Agent Systems, vol. 12, no. 2, pp. 183-198 (2006).

[17] Resnick, P., and Zeckhauser, R. : ”Trust Among Strangers in Internet Transactions: Empirical Analysis of Ebay Reputation System”, The Economics of the Internet and E-Commerce, vol. 11, pp. 127-157 (2002).

[18] Regan, K., Poupart, P., and Cohen, R. : ”A Bayesian Reputation Modeling in E-marketplaces Sensitive to Subjectivity, Deception and Change”, In Proceedings of the 21st National Conference on Artificial Intelligence, Volume 2, pp. 206-212, Boston, MA, USA (2006).

[19] Khosravifar, B., Gomrokchi, M., Bentahar, J., and Thiran, P. : ”A Maintenance-based Trust for Multi-agent Systems”, In Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems, pp.

1017-1024, May 10-15, 2009, Budapest, Hungary (2009).

[20] Zhang, J., and Cohen, R. : ”Evaluating Trustworthiness of Advice About Seller Agents in E-marketplaces : A Personalized Approach”, Electronic Commerce Research and Applications, vol. 7, no. 3, pp. 330-340 (2008).

[21] Fang, H., Zhang, J., Sensoy, M., and Thalmann, N. : ”A Reputation Mechanism for Virtual Reality - Five-sense Oriented Feedback Provision and Subjectivity Alignment”, In Proceedings of the IEEE 10th International Conference in Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 312-319 (2011).

[22] Dropbox, `http://www.dropbox.com`

[23] Google Cloud, `http://www.google.com/cloud`

[24] Wuala, Secure Online Storage, `http://www.wuala.com`

[25] Landers, M., Zhang, H., and Tan, K.L. : ”Peerstore: Better Performance by Relaxing in Peer-to-peer Backup”. In Proceedings of the Fourth International Conference on Peer-to-Peer Computing, pp. 72–79. IEEE Computer Society, Washington, DC, USA (2004).

[26] Kubiatowicz, J., Bindel, D., Chen, Y., Czerwinski, S., Eaton, P., Geels, D.,Gummadi, R., Rhea, S., Weatherspoon, H., Weimer, W., Wells, C., and Zhao, B. : ”Oceanstore: An Architecture for Global-scale Persistent Storage”. SIGPLAN Not, vol. 35, pp. 190–201. ACM, New York, USA (2000).