Perseverance of Uncertainty in Cloud Storage Services through Reputation Based Trust

Vegi Srinivas[1], Vatsavayi Valli Kumari[2], and KVSVN Raju[3]
(Corresponding author: Vegi Srinivas)
Department of Computer Science and Engineering, Dadi Institute of Engineering and Technology[1]
DIET, Anakapalle-531002, Andhra Pradesh, India
(Email: srini.vegi@gmail.com)
Department of Computer Science and Systems Engineering, College of Engineering[2]
Andhra University, Visakhapatnam-530003, Andhra Pradesh, India.
Research and Development Cell, Anil Neerukonda Institute of Technology and Sciences[3]
Sangivalasa-531162, Visakhapatnam, Andhra Pradesh, India.
(Received Feb. 8, 2017; revised and accepted July 13, 2017)

Abstract

The overwhelming technology in the world of computing is Cloud Data Storage and it is also a significant approach to making the highest level of durability, availability, and performance of the services to the users. A sudden and significant change over the data at cloud storage is much more problematic to find the risks. Security plays a vital role in cloud computing and trust is one of the most fascinating and promising factors to prevent uncertainty. Since the organizational hand over direct control over the data, it trusts on the provider to keep that data in a protected way. Clients make sure that the service provider protects data confidentiality by using reputed results to send and storing static data. In this paper, a new approach for reputation based interactions is proposed that are characterized by the trust which is critical for the cloud data persistence and the promise of gaining the advantage in a competitive market.

Keywords: Cloud Computing; Cloud Data Storage; Credibility; Inconsistency; Reputation Based Trust

1 Introduction

The environment of cloud computing offers two basic types of functions: computing and data storage. In the cloud computing environment, consumers of cloud services do not need anything and they can get access to their data and finish their computing tasks just through the Internet connectivity. During the access to the data and computing, the clients do not even know where the data are stored and which machines execute the computing tasks. The data in the cloud storage to be revealed by the user if the provider is considered trustworthy in the field of cloud computing.

There are some questions can be arising in the area of cloud data storage. Where my data is residing in the network? What types of vulnerabilities are exist in Cloud? In the Cloud Data Storage, Data Segregation and Accountability issues are one of the major problems. The cloud storage solution for a specific application or service may change based on many factors, such as Maturity, Performance, Compliance, Risk, Location Demands, Security, Technology Changes, and Changing Business Requirements [5]. To support enterprise customers with a solution flexible enough to meet their application requirements, cloud service providers must offer a broad range of cloud capabilities that falsification the lines between types of cloud infrastructure [11].

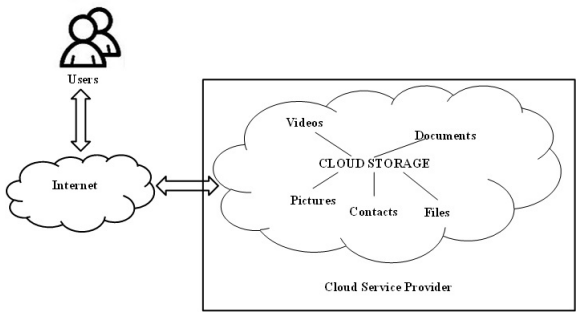


Figure 1: Cloud storage architecture

As shown in Figure 1, the cloud storage architecture will consist of several amenities such as videos, documents, pictures, files, *etc.* The derived trust values or reputation scores must be transparent to and clear enough for the consumers so that they can easily and confidently make the trust-based decision. Users are willing to disclose their data to the cloud provider if the provider

is deemed to be competent, to be of integrity, and to be benevolent. Realizing that trustworthy, high-quality providers will not risk their reputation, users will be less concerned to use the cloud storage service [25].

Cloud computing is one of the emerging fields which replaces the burden of IT industry from spending huge expenditure on resources such as storage and network. Remote storage and easy accessibility of data combined with characteristics such as on-demand self-service, broad network access, resource pooling, rapid elasticity and measured services [24]. Virtualization provides a practical vehicle for transferring compute environments and sharing physical compute resources in the cloud. This approach has been used successfully by financial institutions and the life sciences to solve heavy compute models. It is expensive to run data centers full of servers ready to run complex mathematical models [13].

The computing resources that are provided by cloud service provider's (CSP's) shared to serve all consumers using a multi-tenant form, with different physical and virtual resources dynamically assigned according to consumer demand. The customer generally has no control of the location of the allocated resources. As a result, establishing accountability in distributed and layered architecture is an issue [4].

Trust Foundation has two stage forms; at first, it is trailed by irregular trust upgrade. Next, after the preliminary verification, the security properties of every user established occasionally for conformance with predefined security arrangements. Guarantee the service provider secures data confidentiality by utilizing encryption to transmit information, and utilizing it when putting away static information. The stronger the security, the greater the consumption of computing, memory, and bandwidth resources and the more difficult the service is to use, requiring manual configuration of security mechanism parameters [3].

The management and containment issues with rapid resource pooling are the main drawbacks in the cloud environment. Cloud computing differs from previously studied products and services in the way that it introduces a continuous uncertainty into the relationship between the provider and the user. Although the user depends on the cloud service provider at all time, he has only limited information about the providers's qualities, intentions, and actions [25].

The rest of the paper is organized as follows. In Section 2, we present the trust management and their techniques. The related work is discussed in Section 3. Section 4 describes the reputation based trust models so as to minimize the drawbacks of the existing models and enhance the trust values. The system implementation and experimental results are demonstrated at Section 5, Section 6 shows a small case study on file sharing in the cloud environment and finally; the paper is concluded in Section 7.

2 Trust Management

There are two ways to model the trust or distrust among peers: namely trust and reputation. Trust can be transitive but not necessarily symmetric between two parties. The combined trust model is the combination of three popular models such as identity-based trust, capability-based trust, and behavior-based trust. Lacking trust between service providers and cloud users has delayed the universal acceptance of cloud computing as a service on demand. As a virtual environment, the cloud poses new security threats that are more difficult to contain than traditional client and server configurations. In many cases, one can extend the trust models for P2P networks and grid systems to protect clouds and data centers.

2.1 Trust Management Techniques

Trust can be transitive yet not so much symmetric between two parties. The joined trust model is the blend of three mainstream models, for example, identity-based trust, capability based trust, and behavior based trust. Lacking trust between service provider and cloud consumer has overdue the comprehensive acknowledgment of distributed computing as an administration on interest. Trust management service is a difficult problem due to a unpredictable number of consumers and the highly dynamic nature of the cloud services. The trust management service should be flexible and extremely scalable to be practical in cloud environments.

2.1.1 Trust Models

There are several trust models were already proposed by several researchers, each one having their own advantages and disadvantages. Which models are appropriate based on their security and trust requirements and the systems they need to interface it. Some of the trust models we have discussed in this paper are, Public Key Infrastructure (PKI)-based trust model, Feedback credibility-based trust model, Behavioral-based trust model, Subjective trust model, and Domain-based trust model [10]. Most of the trust models are subject to different kinds of attacks, while a few of them are resistant to particular attacks like false praise or accusation (FPA), Sybil and white washing attacks.

The PKI-based trust model depends on a few leader nodes to secure the whole system. This model may cause uneven load or a single point of failure since it relies on leader nodes too much. Behavioral-based trust model uses history trade records to compute trust. Subjective trust is a personal choice about the definite level of entity's particular characters or behaviors. The Domain-based trust model is mostly used in Grid computing which divides into two kinds of trust; one is in-domain trust relationship and the other is inter-domain trust relationship.

2.1.2 Taxonomy of Trust

A trust metric is a measure of how a member of a group is trusted by the other members of the group. Trust metric can be classified into local trust metric and global trust metric; a local trust metric predicts trust scores that are personalized from the point of view of every single user. On the other hand, a global trust metric computes a single global trust value for every single user. A trust management technique for direct and indirect trust can be calculated and it is defined in [2] as,

$$Direct\,Trust, DT^{A,B} = C^{A,B}(\sum_{i=1}^{p} W_i * T_i^{A,B}) \quad (1)$$

Where $C^{A,B}$ is the confidence factor calculated as a function of collected direct measurements, W_i is the weighting factor for each one of the p event types, $T_i^{A,B}$ is node A trust value of event i regarding node B.

$$Indirect\,Trust, IT^{A,B} = \sum_{j=1}^{n} W(DT^{A,N_j}) DT^{N_j,B} \quad (2)$$

Where n is the number of neighboring nodes A, N_j are the neighboring nodes to A, DT^{A,N_j} is node N_j reputation value of node B, $W(DT^{A,N_j})$ is a weighting factor reflecting node A direct trust value of node N_j.

2.1.3 Trust Evaluation Models

The trust evaluation models are different from the trust models. Firdhous *et al.*, in [7] had discussed about these models: Cuboid Trust, Eigen Trust, Bayesian Network Based Trust Management (BNBTM), GroupRep, AntRep, Semantic Web, Global Trust, Peer Trust, comPrehensive repuTation-based TRust mOdeL (PATROL-F), Trust Evaluation, Time-based Dynamic Trust Model (TDTM), Trust Ant Colony System (TACS).

Cuboid trust represents global reputation trust model which precedes three factors namely, peer's trustworthiness in giving feedback, a contribution of the peer to the system and quality of resources. Eigen trust assigns each peer a unique global trust value in a P2P file sharing network, based on the peer history of upload. BNBTM uses multidimensional applications specific trust values and each domain is evaluated using a single Bayesian network. GroupRep is a group based trust management system.

2.2 Trust Assessment

Integration of security measures, accreditation, bandwidth or customer support are the complex challenges regarding computation of trust. Another issue that is relevant when selecting or designing of trust or reputation mechanism relates to how much customization should be supported and where should be trusted values is aggregated [8]. Trust assessment should be based on not only experiences and user interactions but they also depend on other trustworthy communities.

2.2.1 Feedback

Trust feedback is used for getting the evolution of trust results, depends on the consistency and reliability of the services. The trust system explicitly depends on the credibility of the feedback of the users and their potential behaviors. Cloud consumers either give feedback regarding the trustworthiness of a particular cloud service or request trust assessment for the service. Let j and k be any two peers, then the feedback f about j given by k is represented by f_{jk} and is computed as given below.

$$f_{jk} = \frac{\sum_{i=1}^{n} S_{jk_i}}{t} \quad (3)$$

where t is the total number of transactions performed by k with j. S_{jk_i} represents the satisfaction of k on j in i^{th} transaction and its value is always assumed to be between 0 (not satisfied) and 1 (completely satisfied).

2.2.2 Reputation

Reputation clearly is an important aspect of trust establishment, a fact evident in the numerous reputation-based computational trust models in existence. The quality of the reputation system is primarily indicated by its accuracy and effectiveness in updating periodically. It is the one the important technique in trust because the feedback of the various cloud service consumers give the reputation of the service either positively or negatively.

2.2.3 Quality of Service

The Quality of Service (QoS) evaluation based on recommended trust is about feedback information of clients after executing service. Total set of QoS attributes is Q = T (Execution Time), D (Reliability), U (Availability), H (Throughput), and R (Comprehensive Evaluation) [12]. Eigen Trust algorithm is based on the notion of Transitive Trust. Each peer calculates the local trust value. Trust is stored in opinions, which are a 4-tuple (b,d,u,a): b-belief, d-disbelief, u-uncertainty, a-a-priori trust, where (b+d+u)=1.0 and a=[0, 1]. Eigen Trust requires the inclusion of pre-trusted users to get good performance. The longer time pasts, the more the trust degree reduces.

The importance of Eigen Trust in this paper is to get a global trust value with more weight given to pre-trusted peers. The main advantage of Eigen Trust is scalable computation and the trust does not weaken via transitivity. At the time of joining the new peer in the network and does not so far know anyone; the peer uses the perception of the network provided by the pre-trusted peers, from whom it can learn who else to trust.

2.3 Cloud Trust Models and Their Limitations

In the cloud computing environment, there are several trust models have been defined so far. But, none of these trust models satisfies the qualitative service provided to

the user. So, our approach is to provide a qualitative service to the user from the cloud provider through recommender based trust. Some of the existing trust models in cloud computing are discussed below:

A cloud trust model which holds two layers of trust called inside trust layer and contracted trust layer is proposed by Sato *et al.* in [19]. Both the layers, however, give trust in a layered way yet the trust figured are inside to the affiliation. The Cloud Service Provider (CSP) has nothing to do with the advantages' security. So the affiliation needs to have a private cloud to secure its information which is unfeasible with minimal/medium affiliations.

Shen *et al.* in [20] and Shen and Tong [21] have proposed trusted processing development for trust appraisal. The principal shortcoming of this model is that the fundamental basic arranging relies on upon Trusted Computing Platform [TCP] which is difficult to facilitate with isolated registering concerning equipment.

Alhamad *et al.* in [1] have proposed Service Level Agreement (SLA) based trust show just and no utilization or evaluation has been created or depicted. This model is notoriety based trust that has a disadvantage that customer with high scores for reputation can cheat customer in a couple of trades regardless of the way that they get negative criticism. This model has a concentrated development displaying, so every one of the organizations and reputation information has the single purpose of disappointment.

In Role Based Trust show the trust relies on upon the parts, ID used for TCP, standard confirmation for affirmation. The gear keeps up a specialist key for each machine and it utilizes the master key to delivering unique subkey for every setup of the machine. The data mixed for one setup can't be decoded in another outline of the same machine. In case the machine's outline changes the session key of the adjacent machine won't be significant.

The Active Bundle Scheme [16] proposed in perspective of Identity Management model approach is free of an outcast, it is less disposed to assault as it lessens the threat of association ambushes and side channel assaults, on the other hand, it is slanted to foreswearing of organization as dynamic gathering may in like manner be not executed at all in the remote host.

3 Related Work

Cloud computing services are continually evolving and providers are offering new options, but it is not always in their best interests to enable data mobility. As IT organizations assume a greater role as service broker to the business, they must take ownership of ensuring that technologies from multiple vendors integrate seamlessly [14]. The cloud service provider's reputation reverts the overall view of a community towards that provider; therefore it is more useful for the cloud users (mostly individual users) in choosing a cloud service from many options without particular requirements.

3.1 Sources of Uncertainty

The data in the cloud is not always reliable and it is not under control by the provider. Also, applications that are hosted by the provider may not be available all the time, and/or they may not present the latest versions of these applications. As such, the client may encounter uncertainties with respect to these applications or the results that are delivered by these applications [6]. Most of the existing sources of uncertainty are:

1) Missing information.

2) Trusting the available information.

3) Inconsistency of available information.

4) Irrelevant information.

5) Interpretable information.

3.2 Modeling Uncertainty

There are various qualitative and quantitative approaches to model uncertainty. Uncertainty alone (without the consideration of Trust aspect of the information source) can be modeled as one of the following ways:

1) Probabilistic logic: This is the most common and widely used way of representing uncertainty.

2) Fuzzy logic: This approach allows to classify data into different classes called Fuzzy Sets, depending upon their relevance or closeness to the set.

3) Dempster - Shafer belief theory: It basically deals with measures of two main aspects belief and credibility.

4) Subjective logic: Based on probabilistic logic and Dempster-Shafer evidence Theory (DST), this approach has come up as one of the important ways to model uncertainty.

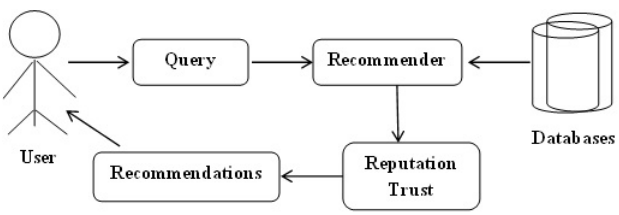


Figure 2: Reputation based trust organizational design

A data source may have different levels of trust at different times and different contexts. Uncertainty on a data source which has provided accurate measurements fairly regularly is less than compared to a new data source [17].

4 Reputation based Trust

Trust and reputation are related but different. Mostly, trust is between two entities; but the reputation of an entity is the collective opinion of a community towards that entity. Usually, an entity that has a high reputation is trusted by many entities in that community; an entity, who needs to make trust judgment on a trustee, may use the reputation to calculate or estimate the trust level of that trustee [9].

The reputation of a cloud service provider follows the overall view of a community against that provider, therefore it is more useful for the cloud users in choosing a cloud service from many options without particular requirements. The user always puts a request(query) to the recommender about the reliable services from the cloud databases and get several recommendations for the services based on trust and reputation as presented in Figure 2.

Trust in cloud computing services is based on several recommendations provided by numerous researchers. Nowadays recommender-based trust models are used in several e-commerce business enterprises, like Amazon and E-Bay. In recommender systems, it is clear based on the other users' ability to provide valuable recommendations. Since the number of direct interactions of the users is very small, so the number of direct relationships plays a minor role in the process of recommendation. The trust relationships between the users are not static but dynamically change over time which may lead to change the recommendation results [26].

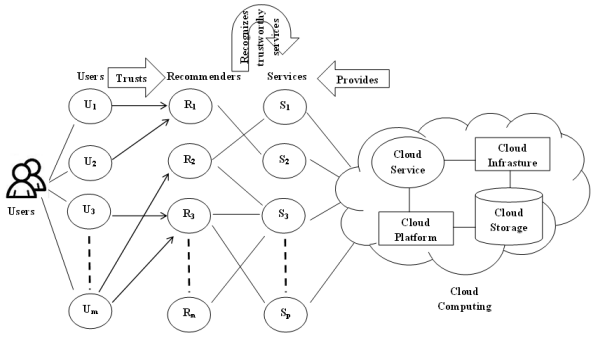


Figure 3: Relationship between users, recommenders and services

In Figure 3, U_1, U_2, U_3, $\cdots$, U_m are the users, R_1, R_2, R_3, $\cdots$, R_n are the recommenders and S_1, S_2, S_3, S_p are the services provided by cloud service providers. Several users can put requests for different services like an infrastructure or software or a database. Each user in this environment gets the feedback from the recommenders on a particular service over a cloud service provider. Talal *et al.*, in [15] discussed trustworthiness of a certain cloud service s, and then the trust result,

$$T_r(s) = \frac{\sum_{l=1}^{|V(s)|} F_c(l,s)}{|V(s)|} \quad (4)$$

Where $V(s)$ is the all trust feedback given to the cloud service $s, |V(s)|$ represents the length of $V(s)$. $F_c(l,s)$ are trust feedbacks from the l^{th} cloud consumer weighted by the credibility. The weights can be calculated based on the consumer experience and satisfaction on the cloud services.

A cloud user is an individual or an organization that has a formal contract or arrangement with a cloud provider to use several resources made available by the cloud provider. Whereas, the cloud provider is an organization which provides cloud-based resources to the consumer. Finally, the recommender is also an individual/organization which analyzes the feedbacks coming from several attributes about the services in the cloud and also recommends to the cloud user whether he/she remain or terminate the services from the cloud providers.

Reputation-based trust is eventually assessed through several trust feedback mechanisms. Each one is having their own advantages and disadvantages during the process of trust evaluation. The users can select a particular service based on their preferences from the cloud service provider; meanwhile, the users can have a direct interaction with the recommenders to get the trust feedback. Non-negative weight is added to the feedback based on recent transactions. We also consider the old transactions so as to give specific weightage to the recommenders to calculate the trust value for the service providers.

5 Implementation and Experimental Results

In this research, a new approach of reputation-based trust evaluation was proposed which is based on weightage given to each and every transaction of the service for the cloud storage to minimize the uncertainty. Our projected trust model helps both the recommender and cloud user, where the user can make a decision on whether to continue or discontinue the service with the service provider.

Trust facilitates users to select the best available service in a diverse cloud infrastructure. Trust value is calculated using three parameters; capability, behavior, and feedback. A more serious type of attack is when malicious peers exploit file sharing networks to distribute viruses and Trojan horses. Peers also need to detect inauthentic file attacks, in which corrupted or blank files are passed off as legitimate files. Before going to undertake a transaction, peers should decide who to trust based on the reputation system which helps to address this need by establishing a trust mechanism [23].

Malicious peers can weaken the reputation system by assigning underprivileged reputation ratings to honest

peers and privileged ratings to other malicious peers. Most of the existing reputation systems absorb into their trust model in view of the correlated trust, to deal with malicious feedback: peers reputed to provide trustworthy service, in general, will likely provide trustworthy feedback.

In Equation (3), an equal importance is given to the satisfaction values due to the most recent transactions as well as the oldest transactions. While in [22] different weights were attached to the satisfaction values, we suggest another addition to the above equation show the difference between the most recently performed the transaction and not so recently performed a transaction. So, that we should have to minimize the responses from malicious peers [18]. Assume int is an interval representing a set of transactions performed during a time period. Let $int = 0$ represent the most recent period and int=1 be the next recent period. Assume the i^{th} transaction was performed in an interval int. Then its corresponding S_{jk_i} is adjusted according to the following equation.

$$f_{jk} = \frac{\sum_{i=1,n\ int=0,|tf|\left[\frac{(ts-2^{int})+1}{ts}\right]} * S_{jk_i}}{n} \quad (5)$$

where ts stands for timestamp which represents the exact time taken when the transaction was performed, tf for timeframe where the considerable past time is categorized into intervals int numbered from 0 to $|tf|$ onwards. Equation (2) allows graceful reduction of feedback ratings as they get old. Figure 2(b) shows how a satisfaction rating fades with time. The significance is that the recent ratings overweigh the past ratings. The advantages are twofold:

1) The recent feedbacks are given more importance and hence;

2) Reputation computation gets more dynamic.

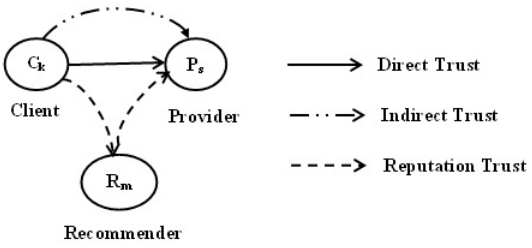


Figure 4: Reputed trust through recommendations

As in Figure 4, C_k is the service request by the customer to the provider through direct interaction, P_s is the various services provided by the CSP to the customers, and R_m is the Recommender used for giving the feedback to the customers about the trusted services. So, the weightage to the specified service for the user is:

$$w = (tv)^p \quad (6)$$

$0 \leq tv \leq 1$, where tv is a single value for local trust which is suggested by the recommender and p is the time period in which the transaction is done between the user and the service provider. The local trust value can be projected based on feedback given by the trustworthy users to the recommenders.

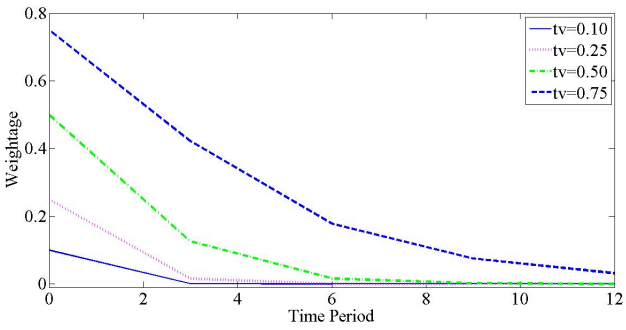


Figure 5: Weightage to the cloud services based on time and trust values

Since, the trust values of the services always lie in between 0 and 1, and then in Fig.5 shows that if the trust value is 0.1, the weightage given to the services is computed based on the time period. For the longest time period and low trust value, the weightage is below 0.1. If the trust value is 0.25 and the weightage is below 0.2. If the trust value is 0.50 and the weightage is below 0.4. Finally, if the trust value is 0.75, then the weightage is nearest to 0.5.

So, now we have to calculate the reputation based trust value for the specified service for a particular user in a specific time period with the given weightage is:

$$RT\,(Q,S) = G_t.\sqrt{(w)} \quad (7)$$

Where, Q is a service requester, S is a service provider, G_t is a global trust value on a particular service and w is weightage which is already computed in the equation 4. Here, the square root is used for increasing the weightage so as to give the preference to the recent transactions. Based on the above equation, we calculate the reputed trust to each and every transaction between the service requester and service provider, in order to minimize the uncertainty about the services.

The algorithms that are used for the above computations are presented as follows.

Algorithm 1 is used to calculate the difference between the most recent transaction and not so recently performed a transaction. Here S is the satisfaction value, N is the number of Common Vendors and f is the feedback.

Algorithm 2 is used to calculate the weightage to the specified service. Here tv is local trust value, w is the weightage and p is time period.

Algorithm 1 Age of transaction

1: Begin
2: Input: S, N Output: f
3: Let ts be the timestamp
4: Let tv is the transaction value
5: Time is categorized into intervals, $int \; \epsilon \; tf$
6: If int is in between 0 and tf, then $tv := tv + \left(ts - 2^{int}\right) + 1/ts, where \; ts \; \epsilon \; tf$
7: Compute the feedback
8: End

Algorithm 2 Weightage

1: Begin
2: Input: tv, p Output: w
3: Let p is the time period
4: Calculate the local trust tv based on feedback f_{jk}
5: Compute the weighting factor $(w) = (tv)^p$
6: End

Algorithm 3 is used to calculate the reputed trust between the service requester and service provider for each and every transaction. Here Q is a service requester, S is service provider, G_t is a global trust value on a particular service.

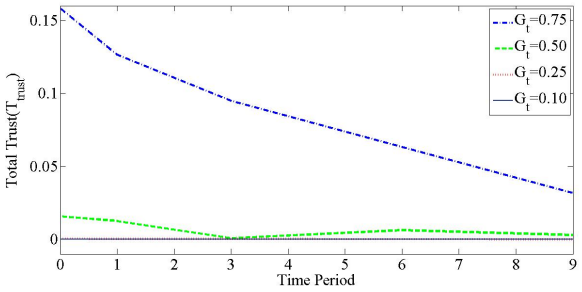
Algorithm 3 Reputed Trust

1: Begin
2: Input: G, w Output: RT
3: Let G is the Global Trust
4: Give more weightage to the recent transactions using $\sqrt{(w)}$
5: Compute the Reputed Trust using $G_t.\sqrt{(w)}$
6: End

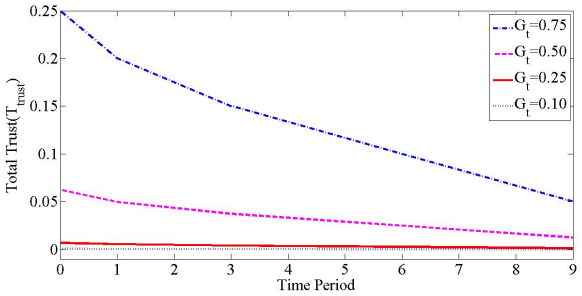
6 Case Study

In this section, a case study related to the distributed file sharing service has been represented under SaaS in a cloud environment and trustworthiness of the related entities have been evaluated based on the proposed trust management model. In the cloud environment, let a specific service of distributing files sharing, where the files have a desired distribution and availability. When any entity wants to share a file in cloud environment then first it needs to ensure that whether a node or entity is trustworthy or not. The trustworthiness can be decided based on service level agreement (SLA) like processing capacity, recovery time, connectivity, peak-load performance, and availability.

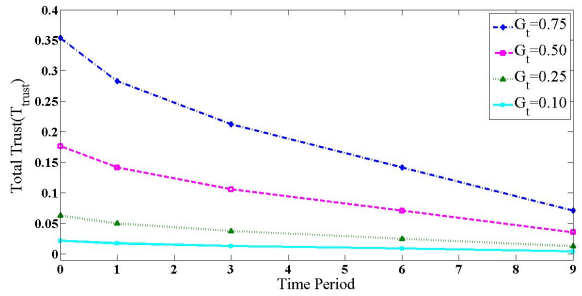
In the Reputed Trust Model (RTM), let the service provider be the vendor v and the trust relationship is established using trust degree based on a request sent to other entities in the cloud. Each entity will maintain two trust tables: direct trust table and the recommended list table. If an entity wants to calculate the trust degree of



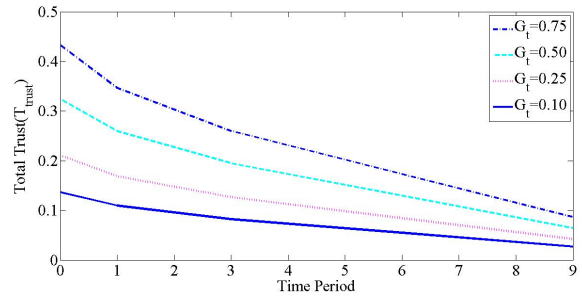
(a) Reputed Trust value to the cloud services based on time period at $tv = 0.10$



(b) Reputed Trust value to the cloud services based on time period at $tv = 0.25$



(c) Reputed Trust value to the cloud services based on time period at $tv = 0.50$



(d) Reputed Trust value to the cloud services based on time period at $tv = 0.75$

Figure 6: Reputed Trust(RT) values to the cloud services

another entity then it first checks the direct trust table. If the trust degree value for the entity exists then it will guarantee for last communication time and then calculate the decay function using Equation (5).

After calculating decay function, reputation based trust for the specified service can be calculated using Equation (7) and where the weightage factor also will be considered. The reputation computation is more dynamic according to decay function effect. Also, the comparative review between the proposed method and the related work is shown in Table 1.

Table 1: Comparative study of proposed method with methods of the related work

Mechanisms	Trust	Reputation	Feedback	Accessibility
Proposed Method	✓	✓	✓	✓
Dipen *et al.*[5]	✓	×	×	×
Ayesha *et al.*[7]	✓	✓	×	×
Shaik *et al.*[8]	✓	✓	×	✓
Firdhous *et al.*[9]	✓	✓	✓	×
Mahbub *et al.*[10]	✓	✓	×	×
Alhamad *et al.*[15]	✓	✓	✓	×
Huang *et al.*[20]	✓	✓	×	×

As shown in Table 1, in most of the related work, just some options in the field of the trusted service description are studied. For example, Dipen *et al.* [6] have considered only the trust, Ayesha *et al.* [8] and Huang *et al.* [23] have considered the trust and reputation. Also, the results show that the provided method acts well than the other related work.

7 Conclusions

- Service availability is one of the significant challenges in the cloud storage to predict the number of requests by several users for the service has to handle at a single point in time. Even though it achieves high availability of services but faces the uncertainties of reliable transactions between the cloud users and providers. Achieving trustworthy services is possible through a reputation-based trust as we are here presented in this research. The mechanism suggested in this paper consider the several communities in general, and allows reputation correction based on the type of community the particular peer belongs to. The simulation results that support our claims have been presented.
- In this research, a number of results can be considered based on the total trust values provided by the recommenders. Even though our proposed system will improve the availability of services by minimizing the malicious peers, but still there is some limitation in our new approach. The proposed system can not be addressed the vendor lock-in; migration of user data and service from one vendor to other is nearly impossible. Future improvements that need to be addressed are how to combine trust and clustering relationships to improve the algorithm performance, and performance of the services in the cloud.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments which helped us to improve the quality and presentation of this paper.

References

[1] M. Alhamad, T. Dillon, E. Chang, "SLA-based trust model for cloud computing," in *13th International Conference on Network-Based Information Systems*, pp. 321-324, 2010.

[2] S. S. Babu, A. Raha, M. K. Naskar, "Trust evaluation based on node's characteristics and neighbouring node's recommendations for WSN," *Journal of Wireless Sensor Network*, pp. 157-172, vol. 6, no. 8, 2014.

[3] J. Chen, Y. Wang, X. Wang, "On-demand security architecture for cloud computing," *Computer*, pp. 73-78, July 2012.

[4] D. Contractor, D. Patel, "Accountability in cloud computing by means of chain of trust," *International Journal of Network Security*, vol. 19, no. 2, pp. 251-259, Mar. 2017. (DOI: 10.6633/IJNS.201703.19(2).10)

[5] Dimension Data, *Service Providers Need Flexible Cloud Services to Compete*, White Paper, 2012. (`www.dimensiondata.com/onecloud`)

[6] B. N. Farah, "A model for managing uncertainty on the cloud," *Journal of Management Policy and Practice*, vol. 14, no. 6, 2013.

[7] M. Firdhous, O. Ghazali, S. Hassan, "Trust management in cloud computing: A critical review," *International Journal on Advances in ICT for Emerging Regions*, pp. 24-36, vol. 4, no. 2, 2011.

[8] S. M. Habib, S. Hauke, S. Ries and M. Muhlhauser, "Trust as a facilitator in cloud computing: A survey," *Journal of Cloud Computing: Advances, Systems and Applications*, pp. 1-19, 2012.

[9] J. Huang, D. M. Nicol, "Trust mechanisms for cloud computing," *Journal of Cloud Computing: Advances, Systems and Applications*, pp. 2-9, 2013.

[10] A. Kanwal, R. Masood, U. E. Ghazia, M. A. Shibli and A. G. Abbasi, "Assessment criteria for trust models in cloud computing," in *IEEE International*

Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, pp. 254-261, 2013.

[11] C. Lan, H. Li, S. Yin, L. Teng, "A new security cloud storage data encryption scheme based on identity proxy re-encryption," *International Journal of Network Security*, vol. 19, no. 5, pp. 804-810, Sep. 2017. (DOI: 10.6633/IJNS.201709.19(5).18)

[12] W. Lu, X. Hu, S. Wang, X. Li, "A multi-criteria QoS-aware trust service composition algorithm in cloud computing environments," *International Journal of Grid and Distributed Computing*, vol. 7, no. 1, pp. 77-88, 2014.

[13] Nasuni, *Understanding Security in Cloud Storage*, White Paper, 2010. (`www.nasuni.com`)

[14] NetApp,*Data Fabric: Realize the Full Potential of the Hybrid Cloud*, White Paper, 2017. (`www.netapp.com/datafabric`)

[15] T. H. Noor, Q. Z. Sheng, "Trust as a service: A framework for trust management in cloud environments," in *Web Information System Engineering (WISE'11)*, pp. 314-321, 2011.

[16] R. Ranchal, B. Bhargava, "Protection of identity information in cloud computing without trusted third party," *29th IEEE International Symposium on Reliable Distributed Systems*, pp. 1060-9857, 2010.

[17] M. Ravi, Y. Demazeau, F. Ramparany, "Managing trust and uncertainty for distributed AI systems," *RJCIA*, 2014. (`https://rjcia2014.greyc.fr/sites/rjcia2014.greyc.fr/files/rjcia2014_submission_3.pdf`)

[18] P. RVVSV, V. Srinivas, V. V. Kumari, R. KVSVN, "An effective calculation of reputation in P2P networks," *Journal of Networks*, vol. 4, no. 5, July, 2009.

[19] H. Sato, A. Kanai, S. Tanimoto, "A cloud trust model in a security aware cloud," *10th IEEE/IPSJ International Symposium on Applications and the Internet (SAINT'10)*, pp. 121-124, 2010.

[20] Z. Shen, L. Li, F. Yan, X. Wu, "Cloud computing system based on trusted computing platform," in *International Conference on Intelligent Computation Technology and Automation (ICICTA'10)*, vol. 1, pp. 942-945, 2010.

[21] Z. Shen, Q. Tong, "The security of cloud computing system enabled by trusted computing technology," in *2nd International Conference on Signal Processing Systems (ICSPS'10)*, vol. 2, pp. 11-15, 2010.

[22] M. Srivatsa, L. Xiong, L. Liu, "Trustguard: Countering vulnerabilities in reputation management for decentralized overlay networks," in *Proceedings of the 14th International Conference of World Wide Web*, pp. 422-431, 2005.

[23] G. Swamynathan, B. Y. Zhao, K. C. Almeroth, "Decoupling service and feedback trust in a Peer-to-Peer reputation system," *International Symposium on Parallel and Distributed Processing and Applications (ISPA'05)*, pp. 82-90, 2005.

[24] D. Thiyagarajan, R. Ganesan, "Cryptographically imposed model for efficient multiple keyword-based search over encrypted data in cloud by secure index using bloom filter and false random bit generator," *International Journal of Network Security*, vol. 19, no. 3, pp .413-420, May 2017. (DOI: 10.6633/IJNS.201703.19(3).10)

[25] M. Trenz, J. C. Huntgeburth, D. J. Veit, "The role of uncertainty in cloud computing continuance: Antecendents, mitigators, and consequences," in *Proceedings of the 21st European Conference on Information Systems (ECIS'13)*, pp. 147, 2013.

[26] J. Yuan, L. Li, "Recommendation based on trust diffusion model," *Research Article, The Scientific World Journal*, June, 2014.

Biography

Vegi Srinivas received his M.Sc. in Computer Science and M.Tech. in Computer Science and Engineering from Andhra University. He is a Research Scholar in the Department of Computer Science & Engineering, JNTUK, Kakinada. He is currently working as an Associate Professor in Dadi Institute of Engineering and Technology, Anakapalli, Visakhapatnam, India. His main areas of interests are Cloud computing, Security, Privacy and Trusted Computing. He is a member of IEEE, ACM, CSTA and Life Member of CSI & ISTE.

Vatsavayi Valli Kumari received her B.E. in Electronics and Communication Engineering and M.Tech. and PhD in Computer Science and Engineering all from Andhra University, India and is currently working as Professor in the same department. Her research interests include Security and Privacy issues in Data Engineering, Network Security and E-Commerce. She is a member of IEEE and ACM and is a fellow of IETE.

KVSVN Raju received the B.E. in Electrical Engineering from Government College of Engineering, Kakinada, India and M.E. in Control Systems from Andhra University, India and obtained the Ph.D. in Computer Science and Technology from IIT, Kharagpur, India. He is currently working as Director, R & D Cell, Anil Neerukonda Institute of Technology & Sciences (ANITS). He is a retired Professor in the Department of Computer Science and Systems Engineering at A.U. College of Engineering, Visakhapatnam, India. His research interests include Data Engineering, Security Engineering and Software Engineering.