Designing a Fuzzy-Logic Based Trust and Reputation Model for Secure Resource Allocation in Cloud Computing

Kamalanathan Chandran, Valarmathy Shanmugasudaram, and Kirubakaran Subramani
Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, India

Abstract: *To plan and improve a fuzzy logic and neural network based trust and reputation model for safe resource allocation in cloud computing is the most important motto of this research. Among the IT professionals in current scenario, the cloud computing is one of the main topics conversed. Now, to revise the security, we employ the trust manager and reputation manager in our proposed approach At first, the user access a resource block through the scheduling manager and a structure will send to the user following accessing the resource block to fill the characteristic values of Trust Factor (TF) and Reputation Factor (RF). The TF and reputation value is after that computed for the resource center and it is specified to the fuzzy logic system and neural network to obtain the Security Score (SS) of a resource center. To offer the security controls is the advantage of our suggested method in accessing the cloud resources from cloud computing owing to different security issues occurred in networks, databases, resource scheduling, transaction management and load balancing.*

Keywords: *TF, RF, fuzzy logic system, SS, resource center.*

Received May 25, 2013; accepted June 19, 2013; published online March 8, 2015

1. Introduction

In current years, cloud computing has become a highlight for the IT specialists due to the potentiality to transform. To execute this technology, immense steps had been taken. To improve this domain [1, 5], the guaranteed profits have found out the companies to spend a big amount of money for research. It is an internet depended service delivery method that presents internet based services, computing and storage for users in all markets that holds financial health care and government. It is attracting massive global investment and this novel economic system for calculating has discovered fertile ground. By issuing services with similar functionality, cloud providers will more and more try to win for customers as the business market is increasing quickly with novel providers entering the market. Based on the offered quality level of those services, there can be massive differences on the other hand. Such an aggressive market requires means to dependably review the quality level of the service providers [7]. In addition, by presenting a variety of computing services cloud computing offers several opportunities for organizations. A lot of researchers pay their consideration to both in the academia and in the industry cloud computing. To work out numerous issues, cloud computing has been broadly employed by both enterprises and individuals [3].

For cloud executions, the cloud computing requires to build up a suitable security [14], even though the benefits of cloud computing is accurate. A main problem that needs particular attention is safety of clouds and the Trust Management (TM) is an important factor for cloud security [4]. In different applications, trust and reputation systems [9] are successfully applied to support the users to identify the dependable and trustworthy providers; for example, eBay, Amazon and application markets for mobile applications. To choose the suitable trustworthy cloud providers, similar methods are necessary to assist the customers. Devoid of bearing in mind other sources and roots of information, existing trust and reputation system depends on customer feedback. Besides, it requires extra parameters [6] that assist the customers in choosing providers in a cloud marketplace. As a result, to assist the customers in making obvious assessments, trust and reputation systems have to progress into the TM system [8] before choosing steady trustworthy cloud providers.

In this document, for secure resource allocation in cloud computing we introduce a fuzzy logic and neural network based trust and reputation model. We employ trust manager and reputation manager to gather the characteristic values for Trust Factor (TF) and Reputation Factor (RF) from the users after employing a resource. At first, users will present a task through the scheduling manager. The scheduling manager after receiving the user's duty, it expresses to the related resource center which the task necessitates to complete. The user requires accessing a resource block in a resource center to execute a task. After carrying out the task, a form will be given to the user to fill the characteristics values for the trust *TF* factor and RF based on the experience. These characteristic values

are applied to work out the security. The *TF* value and RF value for the resource center is calculated based on the characteristic values and given as input to the fuzzy logic system. The fuzzy logic system offer the score value for the resource center based on the *TF* and RF we present as input. From the score value we make a decision whether a resource center is protected or not. The main involvement of our work is as follows:

- We have progressed a mathematical model for computing the *TF* and *RF* based on the characteristic values.
- We have suggested Algorithm 1 i.e., trusts and reputations based Security Score (SS) algorithm.

This paper is arranged as follows: Section 2 demonstrates a few of the associated works, section 3 demonstrates the requirement for security in resource allocation of cloud computing, section 4 describes our suggested method. Section 5 demonstrates our TR-SS algorithm, section 6 deals our experimental results, and section 7 concludes our method.

2. Related Works: A Brief Review

This segment demonstrates a few of the researches seen in the literature for trust based secure model and trust reputation system in cloud computing and grid computing environment. For different distributed system, a trust model has been suggested by Firdhous *et al.* [4]. The TM systems suggested for cloud computing had been examined with particular emphasis on their ability, applicability in sensible heterogonous cloud environment and implementabilty. It was found that not any of the systems was based on solid theoretical foundation during the assessment of those systems and furthermore does not take any superiority of service characteristic for forming the trust scores. Therefore, solid theoretical groundwork for building trust models for cloud computing was necessary. A physical cloud computing security architecture has been proposed by Tripathi and Mishra [14]. Cloud security was turning into a chief differentiator and aggressive edge amid cloud providers. They have discovered the security issues that happen in a cloud computing frame work. It spotlighted on technical security aspect arising from the practice of cloud services and in addition offered a summary of main security aspect related to cloud computing with the outlook of a secure cloud architecture environment. Proactive enterprises and service providers used this security on their cloud infrastructure, to attain security so that they were taken benefit of cloud computing in front of their competitors.

Khan and Hamlen [11] have offered and assessed Hatman: The first full-scale, data-centric, reputation depended TM system for Hadoop clouds. By distinguishing the job replica effects for steadiness, Hatman vigorously estimates node integrity. These capitulated agreement feedbacks for a trust manager rely on Eigen trust. Low overhead and high scalability had been attained by creating both consistency-checking and TM as secure cloud computations; as a result, the cloud's disseminated computing power was influenced to make stronger its protection. For a cloud computing marketplace, a multi-faceted TM system structural design has been proposed by Habib *et al.* [7]. This system offers to identify the trustworthy cloud providers based on different characteristics (e.g., security, performance, compliance) estimated by abundant sources and roots of trust information.

Song *et al*. [12, 13] have suggested that trusted grid computing stress robust resource allotment with security assurance at all resource sites. By lack of security guarantee from isolated resource sites, large-scale grid applications were being holdedup. They have formed a security-binding system through site reputation measurement and trust integration across grid sites. They didn't take care of the *TF* deterministically. As a substitute, they have employed fuzzy theory to deal with the fuzziness or uncertainties following all trust characteristics. The combining was reached by repeated replace of site security information and matchmaking to please user job demands. PKI-based trust system assists grids in multi-site verification and single sign-on operations. On the other hand, cross certificates were not sufficient to review local security conditions at grid sites. For disseminated trust aggregation, they have proposed a fuzzy-logic trust system through fuzzification and integration of security characteristics. They have brought in the trust index of a grid site, which was resolute by site reputation from its track record and self-defense potential attributed to the risk conditions and hardware and software defenses arranged at a grid site.

Vivekananth [15] has suggested that grid system was a vibrant environment where all things shared the resources issued by the other entities. The system permits the synchronized and aggregated apply of geographically disseminated resources, often owned by independent organizations, for working out large-scale issues in science, engineering. Conversely, application composition, resource management and scheduling in those environments were a complex process. Before beginning any transaction, the resource provider as well as the user should be convinced. Mutual trust must be created among the user and the provider. Trust was built on repute. The idea of reputation was eye-catching in peer to peer networks. However, yet it was not flawless in grid computing. They have offered an outline of available reputation based systems for resource selection.

3. Need for Security in Resource Allocation of Cloud Computing

To launch the patterns and trends to improve the quality, there is an essential necessity to securely store, manage, share and examine large amount of complex information. Due to the critical nature of the

applications, it is essential that the clouds to be protected. The main security aspect with the cloud system is that the owner of the data may not have the power to know where the data is situated. The cause is that if one needs utilizing the benefits of cloud computing, one should besides utilize the resource allocation and scheduling presented by clouds. Therefore, we require protecting the data in the middle of untrusted process. The transpiring cloud computing system challenges to deal with the quick growth of web connected tools and supervise large amount of data. Google has now offered the Map Reduce framework for dealing vast quantity of data on commodity hardware. Apache's Hadoop allocated File System (HDFS) is transpiring superior software component for cloud computing combined with incorporated parts such as Map Reduce. The requirement to augment human reasoning interpretation and decision makes abilities that have effected in the emergence of semantic web which is an idea that efforts to convert the web from its current, only human readable form to machine processable form. This in turn has effected in numerous social networking sites with massive amount of information to be shared and managed.

For cloud computing, there are numerous security issues as it covers a lot of technologies including networks, databases, operating systems, virtualization, resource scheduling, transaction management, load balancing, concurrency control and memory management. Therefore, the security issues for some of these systems and technologies are related to cloud computing. For example, the networks that intertwine the systems in a cloud have to be safe. Besides, the virtualization paradigm in cloud computing consequences in a number of security concerns. For design, mapping the virtual machines to the physician machines has to be executed firmly. The protected resource allocation in cloud computing can offer the user to browse firmly. The user can moreover keep their information safe. We come across *SS* of a resource in cloud computing by the user given characteristic value in this job. By this novel method user can judge about a resource center based on *SS* whether it is protected or not.

4. Proposed Resource Allocation in Cloud Computing

Using fuzzy logic and neural network based trust and reputation, this section describes our suggested model for the allocation of resource in cloud computing. The Figure 1 demonstrates a model structure of our suggested model. It contain users, a scheduling manager, a trust manager, a reputation manager and resource centers that has number of resource blocks. The overall procedure is as follows: To access the resource block, the users offer a task which is in the resource center through the scheduling manager. The scheduling manager makes sure that the resource block where it is situated provides the path to the related resource center. The user presents the attributes value for *TF* and *RF* after access the resource block. *TF* value and *RF* value are then given to the fuzzy logic system and then neural network to acquire *SS*.

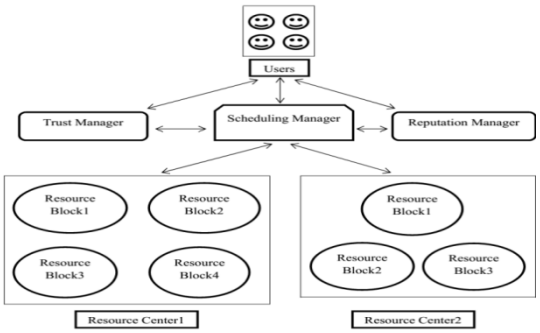

Figure 1. Sample structure of our proposed technique.

4.1. TF of Resource Center

The confident factor of resource center is the total sum of *TF* of every resource block in the resource center. Trust is an important factor in both human society and cyberspace security. Every one of us is conscious of the importance of trusting someone. Since, of the truth that the parameters of the trust are generally personal, the nature of the trust is commonly decentralized. Trust can be named as confidence that a particular party would work in an expected manner in spite of monitoring or controlling the party. Trust is regarded positive and presents good result in uncertain environments. Generally, there has a grey region in expressing the dependability of a computer site [1]. Related to the human relationship, trust is specified by a linguistic term rather numerically. Trust would get fluctuate based on time and environment. Azzedin and Maheswaran [2] presented the definition for the trust and it is as follows: Trust is a strong opinion in proficiency of an entity to act as expected and the strong opinion is not a fixed value related with the entity but rather it is subject to the entity's attitude and employs only inside a particular context at a specified time. The tough opinion can be described as a dynamic value that is found to distance over a set of values vary from very trustworthy to really untrustworthy. Based on the previous experience and is offered for a particular context, the *TF* is formed. The *TF* is based on the specified time instance, as the trust level relating two entities is not significant to be similar for today when compared to a year ago. A few of the characteristics we regarded for the *TF* are as follows:

- Anti-Virus Capability: It is the capability of the resource to guard against viruses and malicious codes.
- Firewall Capability: It is the ability to protect the resource from other network accesses.
- Secured Job Execution: It is the ability of the resources to guarantee the secure implementation of a job.
- Copyright Date: Users trust shopping with repeatedly revised websites. For example, if the

copyright of the website says 2000, then it would be a big red flag for users.
- Corporate Logos: Incorporate the company logos on your website. So that, users will trust your product and services.
- About Us: The about us page demonstrates the detailed history of the company and the user may trust based on the history.
- Privacy Policy: The privacy policy includes a considerable degree of trust since, it shows that you care and respect the customer's personal information.
- Business Address: The business address on webpage demonstrates that you have a bodily location that includes a considerable level of trust among the users.

At first, we have to work out the *TF* of each resource blocks in each resource center. The computation of *TF* of each resource block is as follows:

$$TF_k(rb) = p_a \sum_{i=1}^{m} \left(\sum_{j=1}^{u} \frac{A_{ij} w_i}{TW} \right) \quad (1)$$

Where $TF_k(rb)$: *TF* of resource block, P_a: Probability of users applied the resource block, A_{ij}: *TF* characteristic values given by each user, w_i: Weight value of each *TF* characteristics. *TW*: Total weight value, *m*: Total number of characteristics regarded for *TF*, and *u*: Total number of users.

The equation specified beneath is employed to work out the total weight value of the characteristics. It is the sum of the weight values of every characteristic.

$$TW = \sum_{i=1}^{m} w_i \quad (2)$$

We require to working out the *TF* for the resource center after finding the *TF* value for each resource block in a resource center. For example, in Figure 1, the first resource center has four resource blocks and to locate the *TF* of the first resource center, we ought to know the *TF* value of all the four resource blocks. The *TF* for the resource center is computed as follows:

$$TF(rc) = \sum_{k=1}^{N} TF_k(rb) \quad (3)$$

Where *TF* (*rc*): *TF* of a resource center, and *N*: Total number of resource blocks in a resource center.

4.2. RF of a Resource Center

The reputation mechanism is one of the most important methods which form the basis for the allocated application and system safety, for its improved scalability and litheness. One can trust another on basis of good repute since of the fact. Reputation is described as a measure of trustworthiness in the sense of dependability. For generating trust through social control lacking of trusting third parties, reputation system [11] present a plan. The reputation mechanism offers a plan for generating trust through social control by means of the community based feedback about the past experience of entities. Azzedin and Maheswaran [2] described that reputation of an entity is an anticipation of its attitude based on other entities examinations or information about the entities past attitude at a specified time. A few of the reputation characteristics we regarded for *RF* are as follows:

- *Consistency*: The ability of the resource to execute the anticipated function under stated conditions for a particular period of time.
- *Confidentiality*: The ability of concealing information from unauthorized users.
- *Robustness*: The ability of the system to stay alive from the assaults intended towards that system.
- *Contents Look Current*: If the website is in old format, the users will pay no attention to it. Hence, the website should be brand new with current trends, content and images.
- *Rapid Response*: The fast reply of a webpage will raise the reputation of the webpage since users would like to come to an end their job quickly.
- *Trust Symbols*: By means of trust symbols in the web page demonstrates the users that the web page is guarded against the hackers and viruses.
- *Return Policy*: This really demonstrates that you stand behind you products and it provides the users the comfort of knowing that they can return the product if it is imperfect or they are sad with it.

RF of a resource center is furthermore computed by the similar procedure employed to work out the *TF* of the resource center. The formula to compute *RF* for the resource center is indicated by an equation specified below:

$$RF(rc) = \sum_{k=1}^{N} RF_k(rb) \quad (4)$$

Where *RF*(*rc*): Reputation Factor of a resource center, $RF_k(rb)$: Reputation Factor of each resource block in a resource center, and *N*: Total number of resource blocks in a resource center.

To find *RF* for the resource center, we should to find *RF* for each resource block in the resource center. The formula to assess *RF* for each resource block in a resource center is as follows:

$$RF_k(rb) = P_a \sum_{i=1}^{n} \left(\sum_{j=1}^{u} \frac{B_{ij} l_i}{L} \right) \quad (5)$$

Where P_a: Probability of users applied the resource block, B_{ij}: *RF* characteristics values specified by the user, l_i: Weight value of each *RF* characteristics, *L*: Total weight value of the characteristics regarded for *RF*, *n*: Total number of characteristics regarded for *RF*, and *u*: Total number of users.

The computation of total weight value *L* of the characteristics for the *RF* is specified below.

$$L=\sum_{i=1}^{n} l_i \quad (6)$$

4.3. Fuzzy Logic Model

In our method, this section describes the usage of fuzzy logic system. *TF* value *TF*(*rc*) and *RF* value *RF*(*rc*) are giving as input to the fuzzy logic system to discover *SS* of the resource center. Figure 2 demonstrates the block diagram of the usage of fuzzy logic system in our method.

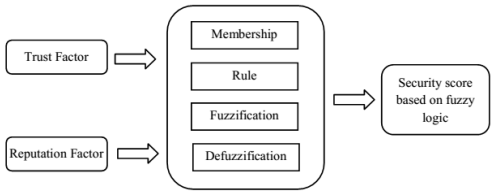

Figure 2. Input and output of fuzzy logic system used in our technique.

The input variables are mapped by set of membership functions in the fuzzy logic system. The act of changing the input value to fuzzy value is called Fuzzification. The Fuzzification in the fuzzy logic system would be based on the rule and the Defuzzification is furthermore based on rule. After Defuzzification we will obtain a single output for the specified number of inputs. In Figure 2 we provide *TF* and *RF* as input to the fuzzy logic system. The fuzzy logic system at first uses the input values of the membership functions and the Fuzzification and Defuzzification will be completed based on the rule. The ultimate output we obtain from the fuzzy logic system is *SS* based on fuzzy logic system. It is signified by an equation beneath:

$$I=TF(rc)*RF(rc) \quad (7)$$

Where *I*: Score we attained as output from fuzzy logic system, *TF*(*rc*): *TF* we present as input to fuzzy logic system, and *RF*(*rc*): *RF* we present as input to fuzzy logic system.

4.4. Neural Network Model

TF and *RF* values are given to the neural network to get *SS*. The neural network learns a predefined set of input/output pairs. Initially, two input neurons are used in the input layer and four neurons in the hidden layer and finally one neuron in the output layer. An input pattern has been applied as a stimulus to the first layer of network units; it is propagated through each upper layer until an output is generated. The output pattern is then compared to the desired output and an error signal is computed for each output unit. The error signals are then transmitted backward from the output layer to each node in the intermediate layer that contributes directly to the output; each unit in the intermediate layer receives only a portion of the total error signal, based roughly on relative contribution the unit made to the original output. This process repeats, layer by layer, until each node in the network has received an error signal that describes relative contribution to the total error. Based on the error signal received, connection weights are then updated by each unit to cause the network to converge toward a state that allows all the training patterns to be encoded. The network trains, the nodes in the intermediate layers organize themselves such that different nodes learn to recognize different features of the total input space. After training, when presented with an arbitrary input pattern that is noisy or incomplete, the units in the hidden layers of the networks will respond with an active output if the new input contains a pattern that resembles the feature the individual units learned to recognize during training. The hidden layer units have a tendency to inhibit their outputs if the input pattern does not contain the feature that they were trained to recognize. The signals propagate through the different layers in the network, the activity pattern present at each upper layer can be thought of as a pattern with features that can be recognized by units in the subsequent layer. The output pattern generated can be thought of as a feature map that provides an indication of the presence or absence of many different feature combinations at the input. The adjustment of the weight values are based on the back propagation algorithm. The neural network eventually gives a *SS* by means of the *TF* value and *RF* value given as input. *SS*s based on the fuzzy logic system and the neural network is merged to get the security for the resource center. *SS* for the resource center is calculated as follows:

$$SS=\frac{\alpha \times I + \beta \times NNS}{2} \quad (8)$$

In the above Equation 8, *SS* is the resource center and *I* is the score obtained based on fuzzy logic system and *NNS* is the score obtained based on neural network and α is the weight value for fuzzy logic based score and β is the weight value for neural network based score.

5. TR-SS Algorithm

This section makes clear about our TR-SS algorithm. TR-SS algorithm is a *TF* and *RF* based *SS* algorithm which applies fuzzy logic system. We are working out the *SS* for a resource center by means of this TR-SS algorithm. A resource center has a number of resource blocks. At first, user presents a task to execute and the scheduling manager directs it to the necessary resource center which has the resource block to execute that task. We want to find the *TF* and *RF* for the resource block by the user specified values after the user applied the resource block.

TF encloses some characteristics and *RF* holds some characteristics. The values for the characteristics are filled by the user after they employed the resource. *TF* and *RF* for the resource block is computed based on the characteristic value and the probability of users employed that resource block. By summing the *TF*

values and *RF* values of the resource blocks in that resource center, *TF* and *RF* for the resource center is computed. *TF* and *RF* of the resource center is next given to the fuzzy logic system and the neural network. The fuzzy logic system fuzzifies the inputs we offer and defuzzify based on the rules and gives a score and the neural network would also give a score. Both the scores from the fuzzy logic system and the neural network are merged and offer the *SS* for the resource center as output.

Algorithm 1: TR-SS.

Input: *Obtain the characteristic values from the users for TF and RF.*
For each resource center rc compute TF and RF.
Work out the probability of number of users Pa employed a resource block RB.
For each resource block in a resource center, compute the trust TF factor.
Do again the fourth steps for next resource blocks in the resource center.
End for
For each resource block in a resource center, compute RF.
Do again the seventh steps for next resource block in the resource center.
End for
Work out the TF for resource center.
Compute RF for resource center.
Replicate the steps from two to eleven for next resource centers.
End for
Present TF and RF of resource center as input to the fuzzy logic system and neural network system.
Fuzzify TF and RF based on rules and defuzzify based on rules.
Merge the scores obtained from fuzzy logic system and neural network.
Output: *SSs for resource center.*

6. Results and Discussions

This section demonstrates the effect of our suggested work. It encloses the experimental setup, fuzzy design result and the presentation study of our method.

6.1. Experimental Setup

With 4GB RAM, our method is executed in java (jdk1.6) that has the system configuration as is processor. We have employed three dissimilar datasets which are financial, medical and RDB for our method. We applied four dissimilar resource centers that have three different resource blocks in our method. The datasets we applied are as resource blocks. With different number of users we examine the presentation of our method because the users will present the attribute values for *TF* and *RF* after they applied the resource.

6.2. Fuzzy System Results

The fuzzy system result is the procedure of input we are giving and the output we get from the fuzzy logic system. Figure 3 demonstrates a model score value we attained from the fuzzy logic system for the membership conditions of *TF* and *RF*.

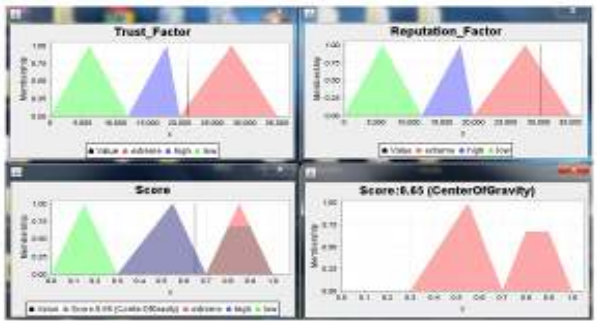

Figure 3. Sample score value obtained from the fuzzy logic system.

6.3. Performance Analysis

This section defines the presentation of our suggested method. To verify the presentation, we use two secure resource centers and two insecure resource centers. Based on the feedback of the users, the first and second resource centers are insecure and the third and fourth resource centers are secure and let us see how our system works. The presentation of our system is verified with dissimilar number of user's. In this part, the representation 'High' in the graph indicates that the resource center is secured and the representation 'Low' in the graph indicates that the resource center is not secured. The clarification of the presentation we attained for our method as shown in the following figures.

Figure 4 demonstrates the presentation of our method based on the feedback of fifty user's by differing the threshold we set after fuzzy logic system together with neural network system to choose whether a resource center is protected or not. Now, when we set the thresholds as 0.2, 0.4 and 0.6, our system demonstrates the complete resource centers we applied as secured and when we set the threshold as 0.8, our system demonstrates the third and fourth resource centers as secured and for the threshold 1, our system demonstrates the complete resource centers we applied for our method is not secured.

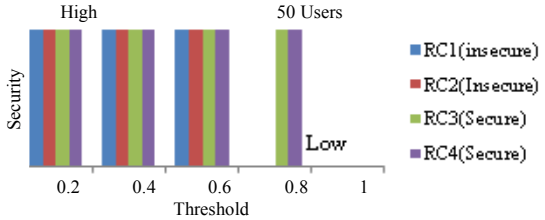

Figure 4. Performance based on feedback of fifty users.

The presentation of our system based on the comment of hundred users is demonstrated in Figure 5 for different threshold values. Now, for the thresholds 0.2 and 0.4, our system demonstrates that the whole resource centers we applied are secure and for the thresholds 0.6, 0.8 and 1, our system demonstrates the resource centers we applied as insecure.

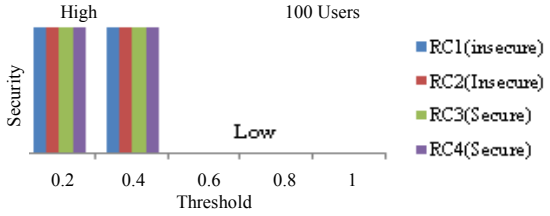

Figure 5. Performance based on feedback of hundred users.

Figure 6 demonstrates the presentation of our system based on the feedback of one hundred and fifty users by varying the threshold which we applied to decide whether a resource center is secured or not. Now, when we place the thresholds as 0.2, 0.4 and 0.6, our system demonstrates the whole resource centers we applied for our method is secured and when the threshold is 0.8, the first and second resource centers are not secured and the third and fourth resource centers are secured. When we place the threshold as 1, our system demonstrates that the whole resource centers we applied as not secured.

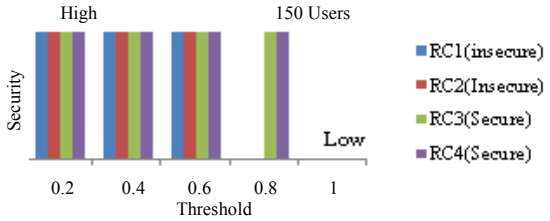

Figure 6. Performance based on feedback of 150 users.

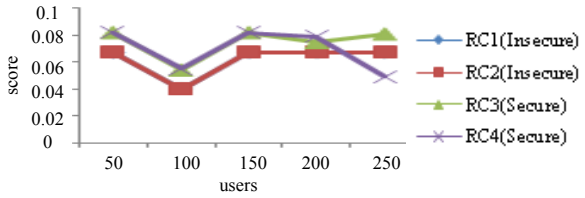

Figure 7. Score values obtained for different resource centers.

The score values we attained from the fuzzy logic system together with neural network for the resource centers we applied in our experimentation are illustrated in Figure 4 for the comments of different numbers of users. At this point, for the comment of fifty users, the score value we attained from the fuzzy logic system together with neural network system for the first resource center is 0.067087044 and for the second resource center is 0.067186321 and for the third resource center is 0.081805798 and for the fourth resource center is 0.081786668. When we reflect on the comment of hundred users, the score values we attained from the proposed system is as follows: 0.040135843 for the first resource center, 0.040135843 for the second resource center, 0.05410748 for the third resource center and 0.054839521 for the fourth resource center. When we reflect on the comment of one hundred and fifty users, the score values we attained for the resource centers we applied are as follows: 0.067146769 for the first resource center, 0.067184832 for the second resource center, 0.081783037 for the third resource center and 0.081755906 for the fourth resource center. When the comment we reflect on for two hundred users, the score value for the first resource center is 0.067161597 and for the second resource center is 0.067174059 and for the third resource center is 0.074504212 and it is 0.079140074 for the fourth resource center. When we reflect on the feedback for two hundred and fifty users, the score values are as follows: 0.067118327 for the first resource center, 0.067153199 for the second resource center, 0.080761123 for the third resource center and 0.049212507 for the fourth resource center.

7. Conclusions

We have suggested a method for secure resource allocation in cloud computing by means of fuzzy logic and neural network based trust and reputation model in this paper. Now, we have applied the trust manager and reputation manager to revise the security of a resource center. At first, user executed a task through the scheduling manager and following the task, user give the characteristic values for *TF* and *RF* of the resource user applied. Based on the characteristics values specified by the users, *TF* and *RF* is computed and specified to the fuzzy logic system and neural network system to discover the *SS* of a resource center. With the comment of dissimilar number of users, we have executed the experimentation of our method and with dissimilar threshold values to make a decision whether a resource center is protected or not.

References

[1] Asma A., Chaurasia M., and Mokhtar H.,"Cloud Computing Security Issues," *International Journal of Application or Innovation in Engineering*, vol. 1, no. 2, pp. 1-5, 2012.

[2] Azzedin F. and Maheswaran M., "Towards Trust-Aware Resource Management in Grid Computing Systems," *in Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*, Washington, pp. 452, 2002.

[3] Chaisiri S., Lee B., and Niyato D., "Optimization of Resource Provisioning Cost in Cloud Computing," *IEEE Transactions on Services Computing*, vol. 5, no. 2, pp. 164-177, 2012.

[4] Firdhous M., Ghazali O., and Hassan S., "Trust Management in Cloud Computing: A Critical Review," *International Journal on Advances in ICT for Emerging Regions*, vol. 4, no. 2, pp. 24-26, 2011.

[5] Gao K., Wang Q., and Xi L., "Reduct Algorithm Based Execution Times Prediction in Knowledge Discovery Cloud Computing Environment," *the International Arab Journal of Information Technology*, vol. 11, no. 3, pp. 268-275, 2013.

[6] Habib S., Ries S., and Muhlhauser M., "Cloud Computing Landscape and Research Challenges Regarding Trust and Reputation," *in Proceedings of Symposia and Workshops on ATC/ UIC*, Xian, pp. 410-415, 2010.

[7] Habib S., Ries S., and Muhlhauser M., "Towards a Trust Management System for Cloud Computing," *in Proceedings of the 10th International Conference on Trust, Security and Privacy in Computing and Communications*, Changsha, pp. 933-939, 2011.

[8] Josang A., Keser C., and Dimitrakos T., "Can We Manage Trust?," *in Proceedings of the 3rd International Conference on Trust Management*, Paris, pp. 93-107, 2005.

[9] Jsang A., Ismail R., and Boyd C., "A Survey of Trust and Reputation Systems for Online Service Provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618-644, 2007.

[10] Khan S. and Hamlen K., "Hatman: Intra-cloud Trust Management for Hadoop," *in Proceedings of the 5th IEEE International Conference on Cloud Computing*, Honolulu, pp. 494-501, 2012.

[11] Malaga R., "Web-Based Reputation Management Systems: Problems and Suggested Solutions," *Journal of Electronic Commerce Research*, Springer Netherlands, vol. 1, no. 4, pp. 403-417, 2001.

[12] Song S. and Hwang K., "Dynamic Grid Security with Trust Integration and Optimized Resource Allocation," *in Proceedings of the International Symposium on High-Performance distributed computing*, Honolulu, pp. 1-16, 2004.

[13] Song S., Hwang K., and Kwok Y., "Trusted Grid Computing with Security Binding and Trust Integration," *Journal of Grid Computing*, vol. 3, no. 1, pp. 53-73, 2005.

[14] Tripathi A. and Mishra A., "Cloud Computing Security Considerations," *IEEE International Conference on Signal Processing, Communications and Computing*, Xi'an, pp. 1-5, 2011.

[15] Vivekananth P., "Trusted Resource Allocation in Grid Computing by using Reputation," *International Journal of Computer Science and Communication*, vol. 1, no. 2, pp. 23-25, 2010.

Kamalanathan Chandran obtained his BE in electronics and communication engineering degree in 2005 from Anna University, Chennai and MTech in applied electronics in 2008 from Dr.MGR University, Chennai. Currently, he is pursuing his PhD degree under Anna University, Chennai. He is presently working as Assistant Professor (Sr.Grade) in the Department of Electronics and Communication Engineering at Bannari Amman Institute of Technology, Sathyamangalam. He is having a total of 7 years of teaching experience in various engineering colleges. His research area includes cloud computing, wired and wireless networks and network security. He is the life member in Indian Society for Technical Education. He has published 7 papers in International Journals, 20 papers in International and National conferences.

Valarmathy Shanmugasudaram received her BE in electronics and communication engineering degree and ME in applied electronics degree from Bharathiar University, Coimbatore in 1989 and 2000 respectively. She received her PhD degree at Anna University, Chennai in the area of Biometrics in 2009. She is presently working as Professor and Head in the Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam. She is having a total of 21 years of teaching experience in various engineering colleges. Her research interest includes biometrics, image processing, soft computing, pattern recognition and neural networks. She is the life member in Indian Society for Technical Education and Member in Institution of Engineers. She has published 38 papers in International and National Journals, 68 papers in International conferences and National Conferences.

Kirubakaran Subramani obtained his BE in electronics and communication engineering from Bharathiyar University, Coimbatore in 2004 and ME in network engineering Anna University of Technology-Coimbatore in 2009. Currently, he is pursuing his PhD degree from Anna University, Chennai in the area of cloud computing. He is presently working as Assistant Professor (Sr.G) in the Department of Electronics and Communication at Bannari Amman Institute of Technology, Sathyamangalam. He is having a total of 6 years of teaching experience in various engineering colleges. His research interest includes P2P networks, overlay networks, data management in cloud computing and distributed systems. He has published 7 papers in International and National Conferences.