Research on Dynamic Trust Management and Evaluation Model in Distributed Environment Based on Cloud Computing Environment

Zhao Li[1, a*]

[1]School of computer science, Wuhan Donghu University, Wuhan, HuBei,430212 China

[a]6115480@qq.com

Keywords: cloud computing; trust model; trust evaluation; trust mechanism; trust management; service selection

Abstract. As a new information service mode and a storing and processing method for mass data, cloud computing brings changes to the age of the Internet service computing; this paper conducts study on the dynamic trust management and evaluation model in the distributed environment based on cloud computing environment.

1. Introduction

As a new information service mode and a storing and processing method for mass data, cloud computing brings tremendous and profound changes to the age of the Internet service computing, which makes huge amounts of computing resources, storage resources, and software resources be provided through Internet platform in a on-demand customized way; therefore, users can enjoy a variety of web services in a more convenient and efficient way. This paper constructs a trust model which can adapt to cloud computing environment and characteristics, and studies the dynamic trust management and evaluation model under the distributed environment in the cloud computing environment.

2. Main trust issues in the cloud computing environment

Due to the huge scale and the unprecedented openness and complexity of cloud computing system, with the expanding in the field of cloud computing in various commercial applications, the security problems face more serious challenges than traditional information system. Based on Internet technologies, cloud computing organize the data center composed of multiple servers, network equipment in a cascade way; the multiple service trust management model in cloud computing environment can more effectively reduce the risk of network and improve the quality of the response. The overall structure of the model is shown in Figure 1.

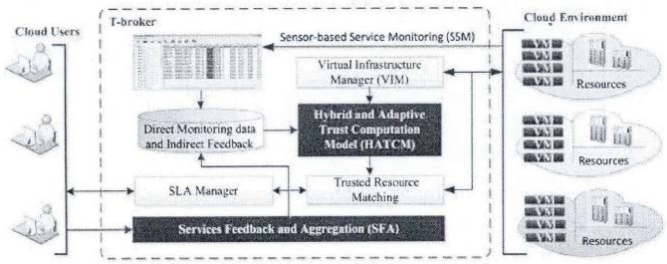

Figure 1 T-Broker model structure

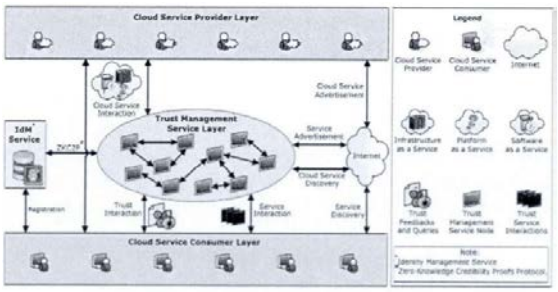

Figure 2 Mixed trust model of T-Broker model

The T-Broker consists of the following important parts: the first one is the service monitor based on sensor; the second one is virtual facility manager; the third one is the service quality manager and trust resource match; the fourth one is the model of the adaptive trust of the sea. As shown in figure 2, HATCM uses a hybrid and adaptive trust model to calculate the global trust of cloud service provider

of , and the trust is a blend of cloud service here real-time performance and the social feedback results of cloud services. Here, HATCM allows consumers to customize the cloud services by customizing their cloud services, according to consumers' business policies and demands. Trust management model based on reputation in the cloud environment: firstly, it puts forward a new protocol that guarantees users' feedback reliability and guarantees user privacy. Secondly, it puts forward a self-adaptive trust measuring mechanism to measure users' feedback, to protect cloud services from malicious user influence and to compare the credibility of different cloud services. Thirdly, it puts forward an available distributed trust model to manage the trust relationship among entities.

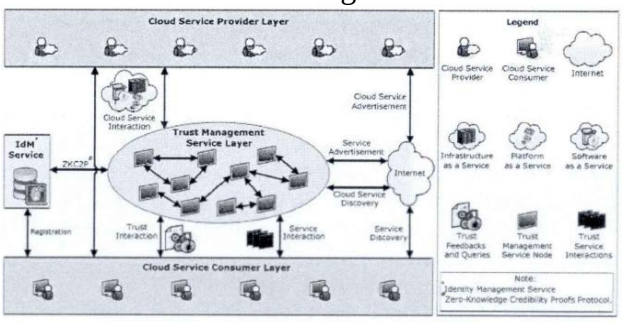

Figure 3 Model structure of CloudArmor□

3. Dynamic trust management and evaluation model analysis in the distributed environment based on the cloud computing environment

Through the analysis of the historical information of interaction among nodes and the comprehensive attainment of the credibility of each node, in each round after the assessment, evaluation results will fail gradually over time. Suppose in a distributed environment, at some assessment time point, there are a total of n computing nodes, respectively written as P 1 ,P 2 , …, P n. Next, we demonstrate the dynamic reliability assessment model by describing the reliability assessment process of other n-1 nodes by describing node P_i ($1 \le i \le n$).

A. Direct credibility assessment

In the direct credibility assessment phase, the computing nodes conduct preliminary assessment on the credibility of each node through the interaction history records with other nodes. Suppose that A_{ij} is the total number of actual direct interaction in a recent time interval between node P_i and node Pj($1 \le j \le n, j \ne i$); T_{ij} refers to the number of successful interaction to node P_i; F_{ij} refers to the number to failing interaction; therefore, A_{ij}-T_{ij}-F_{ij} refers to the number of unidentified successful interaction and failing interaction. Take A_{ij}, T_{ij} and F_{ij}, the basic credibility function from node P_i to node P_j can be built, which is denoted as t_{ij}.We call t_{ij} the direct basic credibility function of node P_i to node P_j. Correspondingly, we call the reliability function as:

$$beI_{ij}((A) = \sum_{B \subseteq A} t_{ij}(B) \ (\forall A \subseteq X)$$

Direct credibility function of node P_i to node P j. As for the other n -1 nodes except for the node itself in the distributed computing environment, each node P_i has a direct basic credibility function t_{ij} ($1 \le j \le n, j \ne i$). the assignment procedure of n -1 direct basic credibility function on the node P_i is as following. Firstly, execute the statement t ij (Φ)=0($1 \le j \le n, j \ne i$) on node P_i ; make the basic credibility number of null set as zero, and the direct credibility can be gained after analyzing historical interaction records through other methods.

$t_{ij}(\Phi)=0, t_{ij}(\{T\})+t_{ij}(\{-T\})+t_{ij}(\{T, -T\})=1$ (1)

The credibility evaluation model can be used to obtain the comprehensive reliability value of the node. The core part of the new model is the comprehensive reliability evaluation stage.

Algorithm 1. Direct credibility assessment method. Process:

Codes for P_i:

for(j|$1 \le j \le n \& j \ne i$){

if(Aij≠0){

$t_{ij}(\{T\})=T_{ij}/A_{ij}$;
$t_{ij}(\{-T\})=F_{ij}/A_{ij}$;
}else{//The total number of historical interaction is 0
$t_{ij}(\{T\})=0$;
$t_{ij}(\{-T\})=0$;
}//endif
$t_{ij}(\{T,-T\})=1-t_{ij}(\{T\})-t_{ij}(\{-T\})$;
}//end for

From the process above, we provide the model structure figure of direct credibility assessment phase structure. Direct credibility assessment phase is mainly to conduct preliminary processing of the original historical interaction records, to calculate the comprehensive reliability.

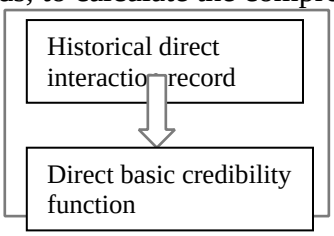

Figure 1 Phase of direct credibility assessment

B. Evaluation of comprehensive credibility

After the broadcast of direct basic credibility function, each node will get the basic and direct credibility function matrix.

$$\begin{bmatrix} t_{12} & t_{13} & \cdots & t_{1n} \\ t_{21} & t_{23} & \cdots & t_{2n} \\ \cdots & \cdots & & \cdots \\ t_{n1} & t_{n2} & \cdots & t_{n(n-1)} \end{bmatrix}_{nx(n-1)} \quad (2)$$

We divide the n(n -1) elements of this matrix into n groups, and each group contains n-1 direct basic credibility functions. Among them, the group j is (t 1j , t 2j , …,t (j -1)j , t (j +1)j , …, t nj)($1 \leq j \leq n$). Here, the basic credibility function of group j represents the initial assessment on P_j from n-1 nodes except for the node P j in distributed computing environment. With the direct basic credibility function t kj provided by node P k ($1 \leq k \leq n, k \neq j$), the corresponding credibility function can be easily got:

$$bel_{ij}((A) = \sum_{B \subseteq A} t_{ij}(B) \ (\forall A \subseteq X)$$

Then, we will demonstrate the process of credibility calculation of node Pj with Pi. When tkj({T,-T})=1.0, the node Pk is unable to determine the success or failure of any interaction with Pj, or the interaction number between node Pk and node is zero, and then, the information provided by node Pk is invalid; our credibility evaluation model will ignore this kind of direct basic credibility function. The value of direct credibility information provided by different nodes has great differences, and we take the value array of ω[n] to represent this kind of difference; at the time of evaluating the credibility of node Pj, the value array is $\omega_j[n]$; (2) The information amount of the direct credibility function belkj provided by the node Pk is directly gained by the Shapley entropy of direct trust function $bel_{kj,}$ and it can also be called as functional factor, to store in v[n]. Algorithm 2 is the computing process of Shapley entropy at the time of recognizing the framework of X={T,-T}. it can be easily seen that, when $bel_{kj}(\{T\})=bel_{kj}(\{-T\})$, the Shapley entropy of reliability function of belkj gets the maximum value; at this time, the maximum entropy of Shapley MAXHY = 1. Considering the node credibility to the continuity of change, we will keep the value distribution in the data set of lastωj[n]($1 \leq j \leq n, j \neq i$) at the end of last round of evaluation, making the credibility factor of this round of evaluation is defined by itself and the direct credibility (see algorithm 3).

Algorithm 2. Calculation of Shapley etropy
functionHY_Shapley(bel_{kj}){//X={T,-T}
ϕ_1=0.5·bel_{kj}({T})+0.5·[1-bel_{kj}({-T})];
ϕ_2=0.5·bel_{kj}({-T})+0.5·[1-bel_{kj}({T})];
HY=-Y1·$\log_2$Y1-Y2·$\log_2 Y_2$;
returnH_Y;
}//HY_Shapley
Algorithm 3: The assignment procedure of weight array omega [n]
function get_weight(n, j){
// n is the number of nodes, returning to the weight array corresponding to the node P j
// last_ω $_j$ [n] is the weight of Pj at the end of the last evaluation
/ / value array
// u [n] is the credit factor of each node
// last u [n] is the credit factor for each node at the end of the last evaluation
// v [n] the utility factor of each node
// αand β are parameters that are introduced to account for the continuity of credibility

// 0<α, β <1
ω j [n] →last ω j [n] ;
u[n] →last u[n] ;
u[i] →1;// Node i Let his own credit factor is 1

v[i] →1-HY Shapley(bel $_{ij}$);
sum→u[i] ·v[i] ;
for(k $_1$≤k ≤n&k ≠i &k ≠j &t k_j ({T , -T})≠1){
u[k] →β·u[k] +(1-β)· t $_{ik}$ ({T});
v[k] →1-HY Shapley(bel $_{kj}$);
sum →sum+u[k] · v[k] ;
}// endfor
for (k 1≤k ≤n&k ≠j &t $_{kj}$ ({T , -T})≠1)
ω j [k] →α· ω j [k] +(1-α)·(u[k] · v[k]/ sum);
// endfor
Last ω j [n] →ω j [n] / retention of this weight is used for the next round of evaluation
Last u [n]-- u [n];// reserve credit factors for the next round of evaluation
Rreturn ω j [n];// returns an array of weights corresponding to node P j
}// get weight
Algorithm 3 is the assignment process of the weight array ω j [n], and it is easy to find that after each successive round of algorithm 3

$$\sum_{\substack{1\le k\le n,k\ne j\\ tkj(\{T,-T\})\ne 1}} \varpi_j[k] = 1$$

Here, we give a brief proof of equation (3), which is known by algorithm 3

$$\sum_{\substack{1\le k\le n,k\ne j\\ tkj(\{T,-T\})\ne 1}} \varpi_j[k] = \sum_{\substack{1\le k\le n,k\ne j\\ tkj(\{T,-T\})\ne 1}} (\alpha \circ last - \varpi_j[k] + (1-\alpha)\circ(u[k]^{\circ}v[k]/sum))$$

$$= \alpha \circ \sum_{\substack{1\le k\le n,k\ne j\\ tkj(\{T,-T\})\ne 1}} last - \varpi_j[k] + (1-\alpha)\circ \sum_{\substack{1\le k\le n,k\ne j\\ tkj(\{T,-T\})\ne 1}} (u[k]^{\circ}v[k]/sum))$$

=α+(1-α)=1.

Therefore, $\sum_{\substack{1\le k\le n,k\ne j\\ tkj(\{T,-T\})\ne 1}} \varpi_j[k] = 1$ is founded.

The new nodes which join in the distributed computing environment are considered separately. The new nodes refer to those which have just joined the distributed computing environment but not to interact with other node node. In the process of the above calculation, if the node P j is new to compute nodes, we will get $ct_j(\{T\})=ct_j(\{-T\})=0$. This will lead to the result that the newly joined nodes cannot get assigned task because of low credibility; we assign certain initial credibility fore new.

4. Conclusions

Cloud computing encapsulate the computing resources, storage resources, software resources and so on in the network and transform them into services, forming a huge shared virtual resource pool; using the fuzzy clustering method based on individual preference, this paper proposes a cloud computing environment service selection model based on trust and personal preference.

Acknowledgements

This paper is supported by Youth Foundation of Wuhan Donghu University.

References

[1] Wang Tao, Zhang Wenbo, Xu Jiwei, Wei Jun, Zhong Hua. Research on Distributed Software System Fault Detection Based on Statistic Monitoring in Cloud Environment [J]. Computer Journal, 2017,02:397-413.

[2] Jin Yu, Wang Fan, Zhao Hongwu, Deng Li. Review on Trust Mechanism in Cloud Environment [J]. Small-scale Microcomputer Systems,2016,01:1-11.

[3] Lin Weiwei, Wu Wentai. Energy-consumption and Management Method Facing Cloud Computing Environment [J]. Software Journal,2016,04:1026-1041.

[4] Shu Jian, Liang Changyong. Cloud Security Trust Model of Multi-source Evidence Fusion Based on DS theory. [J]. Computer Science,2016,08:105-109.

[5] Wang Guiling, Han Yanbo, Zhang Zhongmei, Zhu Meiling. Integration and Service of Streaming Data Based on Cloud Computing[J]. Computer Journal,2017,01:107-125.