Design of frame work for cloud provider's selection based on user requirement and trust

[1]Devi. R,[2]Shanmugalakshmi. R
[1]Assistant Professor,[2]Associate Professor
[1]Department of Information Technology,
[1]Government College of Technology, Coimbatore,India

***Abstract*—Cloud plays a vital role for sharing information or data all over the world. Numbers of users are using the cloud environment for their business or research. Plenty of cloud service providers are available on the cloud environment. Choosing of providers based on their trust and user needs is a difficult task nowadays. Hence in this work a novel idea is implemented to select trust based service provider. This frame work is designed based on competence between the providers and trust calculated based on the customer's feedback.**

***Keywords:* cloud customer, service provider selection, competence, trust, ranking.**

I. INTRODUCTION

The NIST(National Institute of Standards and Technology) specifies the important characteristic of cloud computing as On-demand self-service where A customer can unilaterally provided computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider [1]. According to customer requirements of security and availability, network providers can make various transport routes available [2].

A *Cloud provider* can be an organization orcompany which provides some sort of services based on demand basis. The services can be SaaS – Software as a Service, PaaS – Platform as a service and IaaS – Infrastructure as a Service.

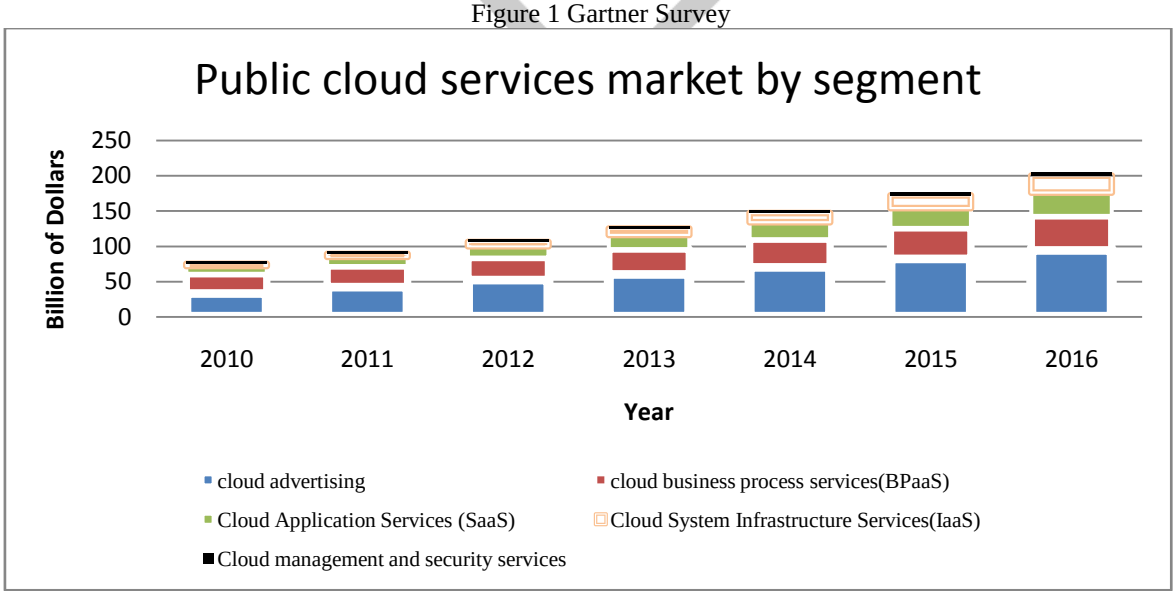
A C*loud customer* can be an organization or individual user who utilizes the service provided by the cloud providers and pay for the utilization.

A *service-level agreement (SLA*) is a contract between a network service provider and a customer that specifies, usually in measurable terms, what services the network service provider will furnish. Many Internet service providers (ISP)s provide their customers with an SLA [3].

With the emergence of new Cloud Providers, identifying one that best suits the business needs of an enterprise is a challenging and difficult task. Adopting a Cloud Provider requires a detailed study of parameters like data security, SLA's and options that address the reduction of capital expenditure [4]. Hence making comparison between the providers for the user needs and arranging them in order based on the trust of providers is needed.

The following modules explain the algorithm and mathematical model of comparison between providers based on a service and ranking them. The importance of the paper is based on the risk calculation of the providers, where in the current situation very less amount of consideration is given for trust and security calculation. This is proved by Gartner survey on February 2013, which is shown as below.

Figure 1 Gartner Survey



This insists the importance of trust and the calculation of trust can be of any number of qualitative and quantitative parameters. In this paper parameters like availability and feedback from users are taken into account for trust calculation. The ranking is calculated based on the indirect tool as feedbacks from the customers after utilization of the service provided by the provider. In paper [5] there are two types of QoS requirements like functional and non-functional parameters that are used for trust calculation. Some of the parameters cannot be measured easily given the nature of the Clouds. Attributes like security and user experience are not even easy to quantify. Here availability is considered as functional parameter and feedbacks are taken as nonfunctional parameter.

There are two main works are consider in this paper to select the best provider. The rest of the paper is organized in the following way, 1) comparing and listing the providers based on providers' availability 2) ranking of providers based on their feedbacks from the registered users.

II. COMPARING AND LISTING OF PROVIDERS

Consider a customer need for a service S1. He searches the internet and found 15 providers provide the service S1. He cannot spend more time to read the entire SLA's provided by the 15 providers. Hence it is a time consuming process and the customer has to compare between the providers manually. Comparison between the providers and ranking the providers based on their trust is the frame work model proposed in this paper.

In [6], an approach is proposed to select software as a service product. Analytic Hierarchy Process (AHP) technique is used for prioritizing the SaaS product features. Three-part methodology is adopted. The first part covers the prioritization of parameters, second part is about product comparison and the third part combines the results obtained from first two parts to rank the products.

From the SLA's of different providers based on the particular service parameters like availability and cost are taken into account. ICMP and HTTP probes are also used for availability calculations. The importance of availability is described in [7] as the cloud grows in importance, it will host more applications and services from the small (such as new and developing web applications) to the large (Amazon, Netflix, etc.). As the users depend on them more and more, services that are run in the cloud need to be highly available.

With SLA's alone calculating availability will not give the availability value. Considering the SLA's as static data for availability and calculating the availability dynamically by using the ICMP and HTTP probes. Set of HTTP probes/ ICMP probes are sent to the provider to calculate the probability of an outage caused by random packet loss. Using the function of packet loss rate and assume k tries for each probe and declare the service down when all tries fail. For packet loss p, the model loss of the request or response [7]:

$$\Pr(\text{outage} \mid k \text{ probes}) = (p + (1 - p)p)^k \quad (1)$$

The following algorithm describes the working principles of listing of providers based on the two parameters availability and cost.

II.1 Algorithm for availability Calculation

- Registration (Customer/ Provider)
  - If (new Customer/ Provider)
    - Add to List
  - Else
    - Update the Details
    - Provider Selection Requirement
- Service Provider Selection based on Competence
  - Let there are p1, p2..., pZ Service providers registered (Z is number of service providers).
  - Let there are c1, c2..., cq cloud Customers registered (q is number of cloud customer).
  - if (customer needs service x)
    - Identify the providers that provide the service x {px1, px2..., pxn}
    - In the set {px1, px2..., pxn}, list the providers based on availability.
- Average K = {A(csp1), A(csp2)..,A(cspn)} [where n>=0, availability parameter is acquired from SLA's of providers].
  - i) M ={csp1, csp2.., cspm} (A(csp)> =K, m<=n);
  - ii) Pr (outage | k probes) = $(p + (1 - p)p)^k$ [where k= no. Of tries for each probe, p= packet loss, Pr = probability of an outage caused by random packet loss]
  - iii) Select E = first 5 maximum of(M)
    Select F =first 5 minimum of(Pr)
  - iv) For each E = F then

$$A = E \cup F \quad (2)$$

III. RANKING OF PROVIDERS BASED ON FEED BACK OF USERS

Evaluating the feedbacks of the customers belonging to a service gives approximation trust calculation of the providers providing that service. Hence trust plays a vital role in ranking the providers

In [6] the metrics are classified as application dependent metrics and user depended metrics. For existing providers, an Adaptive Fused Computing approach is used for dynamic service providers which are based on Information Entropy Theory (IET). For newly registered providers, First Service Last Audit (FSLA) mechanism is used to overcome the trust initialization problem. The architecture is implemented using Eucalyptus framework and the KVM hypervisor.

In this paper the trust is calculated by taking the feedback parameter into consideration and an optimized binary tree is constructed for each providers those are providing the same service. The average level of the optimized trees of the providers is taken as the threshold value. Each and every provider is verified against the threshold value and they are ranked. The following algorithm shows the trust calculation based on feedback using optimized binary tree.

III.1 Algorithm for Ranking the Providers

- Identified providers from 2.1 algorithm, they are taken as input for this algorithm.
- Let the selected providers be {px1, px2, px3...pxn}(where n is the number providers selected for the service x)
- Feedback calculation

  Get the options from the users whenever they are utilizing the service x.
  The options are taken as good, neutral and poor.
  An OBT – Optimized Binary Tree is constructed based on the feedback option of px1 as follows
  If good or neutral a node is created and inserted into OBT of px1
  Else delete a node from px1
  Repeat the above steps until OBT's are constructed for the selected providers
  Let the constructed trees are {OBT(px1), OBT(px2), OBT(px3)...OBT(pxn)}
- Threshold T = Average level{OBT(px1), OBT(px2), OBT(px3)...OBT(pxn)} (3)

  Compare the threshold value against {OBT(px1), OBT(px2), OBT(px3)...OBT(pxn)}
  If OBT(px1) > T then select px1
  Else deselect px1
  Repeat the step 4.2 until 'n' number of
  Providers.
- Let the selected list of providers are { px1, px2, px3... pxt} (where t is number of providers whose level value greater than T)

IV. IMPLEMENTATION RESULTS

IV.1 Availability implementation

The static availability values are gathered from the SLA's provided by the corresponding providers. For example Amazon's availability is taken from the SLA of Amazon [8] for EC2 service is given below.

Table 1 – SLA for Amazon EC2

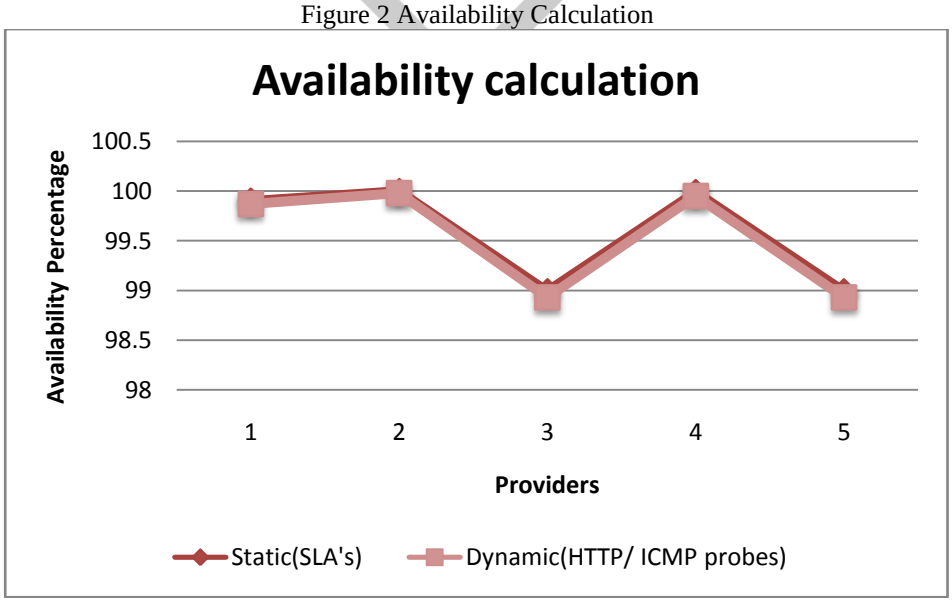
Monthly Uptime %	**Service Credit %**
Less than 99.95% but equal to or greater than 99.0%	10%
Less than 99.0%	30%

From the above table 1 the maximum availability and minimum availability are taken as 99.95 & 99.0

Table 2 – Static and Dynamic calculations

PROVIDERS / METHODS	P1	P2	P3	P4	P5
Static(SLA's)	99.9	99.999	99	99.99	98.99
Dynamic(HTTP/ICMP probes)	99.9	99.998	98.9	99.95	98.93
probability of outage	0.00	0.01	0.01	0.04	0.06

Figure 2 Availability Calculation



From Figure 2, static availability is calculated by obtaining the SLA of the providers for the selected service. Table 1 shows the availability of provider Amazon for EC2 service. Dynamic availability is calculated by sending the probes like (ICMP/ HTTP) to the corresponding service provided website. ICMP probes access the front end information of the web page while http probes are used as rear end (i.e) the probe reaches up to the application level. Based on the probes outage the dynamic availability value is calculated.

IV.2 Feedback Implementation

Whenever the customers utilize the services feedbacks are collected from them. The feedback consists of three options as Good, neutral and poor. Based on their views they rate the providers as good/neural/poor. With this feedback, union of good and neutral is taken as positive feedback.

$$\text{Good U Neutral} = \text{Positive} \quad (4)$$

Similarly poor option is directly proportional to negative

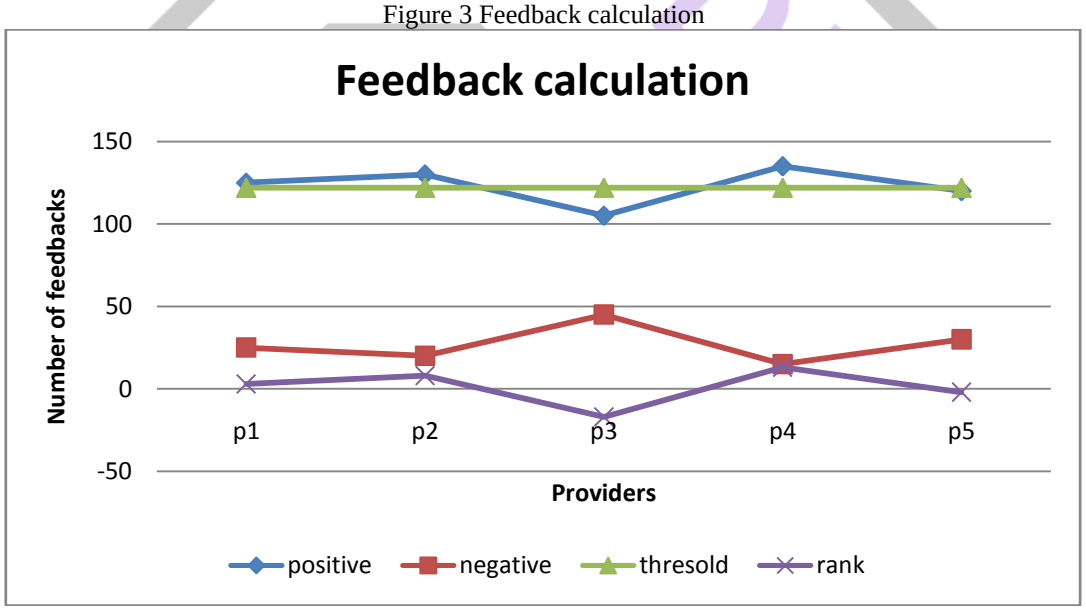
$$\text{Poor} = \text{Negative} \quad (5)$$

Threshold value of positive feedback is taken using Optimized binary tree algorithm and finds the providers greater than the threshold value. Highest positive value will be given first rank and it goes in descending order. Table 2 and figure 2 shows the results of feedback analysis.

Table 3 Raking based on feedbacks

	P1	P2	P3	P4	P5
Positive	125	130	105	135	120
Negative	25	20	45	15	30
Threshold	122	122	122	122	122
Rank	3	8	-17	13	-2

Figure 3 Feedback calculation



From Figure 3, the threshold level is calculated as 122.

The level of 122 nodes represented in an optimized binary tree is selected as the threshold level. The provider's positive value is calculated using equation (4) and the provider with maximum rank value will be ranked first. Which is calculated as follows

$$\text{Rank} = (4) - \text{Threshold} \quad (6)$$

Rank the providers based on decreasing rank values. Example in table 4.3 provider (P4) with rank value 13 will be in the first position then P2, P1, P5, P3 which is the ranking order of the given data set.

V CONCLUSION

Cloud services are utilized enormously now a days, hence choosing a trustworthy service provider is tough task for cloud customers. Nowadays CSPs cloud services are rendered by investing customer preference to provide the best service. To choose an ideal (trustworthy & competence) service, parameters like availability (direct tool) and feedbacks (indirect tool) from users are analysed using this selection framework. Based on the service (IaaS, PaaS, Saas) needed by the customer (Business/Individual), the evaluation method will vary. The existing models are used to calculate the trust and performance of the service providers.

Based on the frame work, we can conclude that, four main steps are needed to evaluate the cloud service providers. (1) Identify the parameters (qualitative and quantitative – availability and feedbacks) (2) Provide values to the qualitative factors (3) Evaluate trust, competence and risk using mathematical calculation (4) Filter and rank the providers based on calculation. Hence the parameters, values, will differ from one cloud customer to another cloud customer. So a framework for finding the risk on the service provider along with the trustworthiness, competence and quantitative performance factors are also needed to provide a best guidance for customer to choose the service provider.

REFERENCE

[1] Michael Hogan, Fang Liu, Annie Sokol and Jin Tong, "NIST Cloud ComputingStandards Roadmap", Computer Security Division, Information Technology Laboratory, NIST, U. S. Department of Commerce, Special Publication 500-291, Jul. 2011.

[2] White Paper "Cloud Computing - Alternative sourcing strategy for business ICT." Published by T-Systems Enterprise Services GmbH , Germany

[3] White paper "Comparing Public Cloud -Service Level Agreements" (Questions to ask when evaluating public cloud SLAs) .

[4] White paper "Cloud Computing Services–A comparison" by Torry Harris.

[5] Saurabh Kumar Garg , Steve Versteeg and RajkumarBuyya, "SMICloud: A Framework for Comparing and Ranking Cloud Services", 2011 Fourth IEEE International Conference on Utility and Cloud Computing.

[6] Xiaoyong Li, Huadong Ma, Feng Zhou, and XiaolinGui, "Service Operator-Aware Trust Scheme for Resource Matchmaking across Multiple clouds", IEEE transactions on parallel and distributed systems, vol. 26, no. 5, may 2015.

[7] Zi Hu1, Liang Zhu, Calvin Ardi , Ethan Katz-Bassett, Harsha V. Madhyastha , John Heidemann , and Minlan Yu "The Need for End-to-End Evaluation of Cloud Availability" Springer International Publishing Switzerland 2014 PAM 2014, LNCS 8362, pp. 119–130, 2014.

[8] Web site: https://aws.amazon.com/ec2/sla/