UPDATING TRUST VALUE IN CLOUD FEDERATION USING ITERATIVE APPROACH

By

Asiya Mushtaq

A thesis submitted to the faculty of Information Security Department, Military College of Signals, National University of Sciences and Technology, Islamabad in partial fulfillment of the requirements for the degree of MS in Information Security

August 2018

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis written by Ms **Asiya Mushtaq,** Registration No. **NUST201463796MMCS25214F**, of **Military College of Signals** has been vetted by undersigned, found complete in all respect as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial, fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the student have been also incorporated in the said thesis.

Signature: ___________________________________

Name of Supervisor: Asst Prof Dr. Rabia Latif, PhD

Date:______________________________________

Signature (HoD):__________________________

Date: __________________________

Signature (Dean/Principal): ___________________

Date: ____________________

ABSTRACT

Cloud computing has emerged as a powerful technology over the past few years to aid quick and low-cost access to IT resources. Computing power, storage, applications, development platforms and infrastructure facilities are readily available to the consumers on pay-as-you-go basis irrespective of their geographical location. Due to an immense growth in cloud computing technology, much of the user base and organizations are shifting to cloud environment. Similarly, to meet the requirements of cloud consumers, cloud services providers (CSPs) too collaborate among each other and distribute the load and infrastructure to provide seamless services to the user. Instead of making new resources available to the consumer, the CSP reduces this cost by hiring them from the other CSP. On other hand, the underutilized resources of the collaborating CSP also produces revenue for it. This concept is acknowledged as Cloud Federation. Security and privacy issues become a hindrance in cloud federation. Thus a need for establishing and maintaining trust factor between the CSPs is essential. Existing research has developed much in the field of dynamically evaluating trust value. However, a mechanism still needs to be worked out to keep the trust value up-to-date for a new CSP entering in the cloud federation. Therefore, the focus of this research is to propose an enhanced technique to evaluate trust values by considering more SLA parameters then existing. The trust value thus established, will be updated on regular intervals of time in an iterative manner.

DECLARATION

I hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification either at this institution or elsewhere.

(Asiya Mushtaq)

DEDICATION

This thesis is dedicated to

MY BELOVED PARENTS
AND
HUSBAND

for their love, endless support and encouragement.

ACKNOWLEDGEMENTS

I would thank Allah Almighty for endowing me with His countless blessings. I express my appreciation to my colleagues and the faculty for providing their enormous support to help me in this research. Without their relentless backing, motivation, and prayers, I would not have reached the culmination point.

I extend my deepest gratitude to my supervisor; Assistant Professor Dr. Rabia Latif, who provided me tremendous support and encouragement for the successful completion of this tedious task.

Finally, I am grateful and thankful to Military College of Signals and National University of Sciences and Technology (NUST) for providing me the platform and the resources to achieve excellence.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

CC	Cloud Computing
CSP	Cloud Service Provider
IaaS	Infrastructure as a Service
IT	Information Technology
NIST	National Institute of Science and Technology
PaaS	Platform as a Service
KPI	Key Process Indicators
PC	Personal Computer
QoS	Quality of Service
SaaS	Software as a Service
SLA	Service Level Agreement

Chapter 1

INTRODUCTION

1.1 Introduction

This chapter provides an overview of the importance of Cloud Computing in today's world of IT followed by various issues with a special focus on trust in CC technology by the end-users and consumer parties. The chapter also highlights the motivation for carrying out research in this direction along with its objectives. This chapter concludes with the organization and structure of thesis.

Cloud Computing has emerged as a powerful technology and has interestingly established its roots not only in large scale organizations but also in small and medium sized organizations owing to its wide range of advantages. The technology has been quite successful in efficient and effective utilization of shared resources (computing power, infrastructure, services, storage etc.) while remaining cost-friendly and highly available to the consumers. Cloud computing works on on-demand and pay-per-use mechanism.

Contrarily, Cloud computing is also facing numerous challenges like privacy concerns, protection of data from unauthorized access, backup and recovery plans, availability, integrity and cost management capabilities. Trust in digital world is just like the conventional trust in our social life. Level of trust in daily life is affected by new experiences and changes with the circumstances we face, it is proportional to the level of coordination, good relationships and mutual cooperation between the two. With the advent of IT, the design and implementation of secure information systems relies on trust factor. Legal frameworks had been proposed and implemented at various levels to establish a trusted relationship between business entities for financial transactions.

In cloud computing environment, trust is symbolic of the QoS provided by the Cloud Service Provider with respect to the services offered. Trust is calculated by using various trust mechanism proposed by researchers. Trust management cycle comprises of trust establishment, its renewal methods and mechanism to withdraw trust. Trust management in cloud computing different from conventional trust because of inherent characteristics of cloud computing, for example location, userbase, QoS parameters, online intrusion etc.[1]

Building trust in CC is a challenging job and reserves a prime importance because all the business companies and organizations reside their confidential business data over the cloud. The blind storage location of data is invisible to the organization. Therefore, it is necessary to take enhanced measures to ensure data security in best possible so as to gain the confidence and eventually the trust of cloud consumers in a a certain CSP. According to a survey by Fujitsu Corporation [2], 88% of the customers across the globe are highly concerned about the privacy of their data in cloud. An environment with enhanced trusted and data security will attract more customers.

Cloud computing now-a-days has evolved into a new concept i.e Cloud Federation to facilitate the ever growing requirements of the customers. The versatility of the nature of requirements of an organization is such that a single CSP at times can not come upto the mark and therefore it collaborates with another CSP over some terms and agreements to meet the demands of the customer. The infrastructure though seems one to the customer but at the backend it is virtualized and distributed onto multiple CSPs. The basic concept of cloud federation is represented in Fig 1

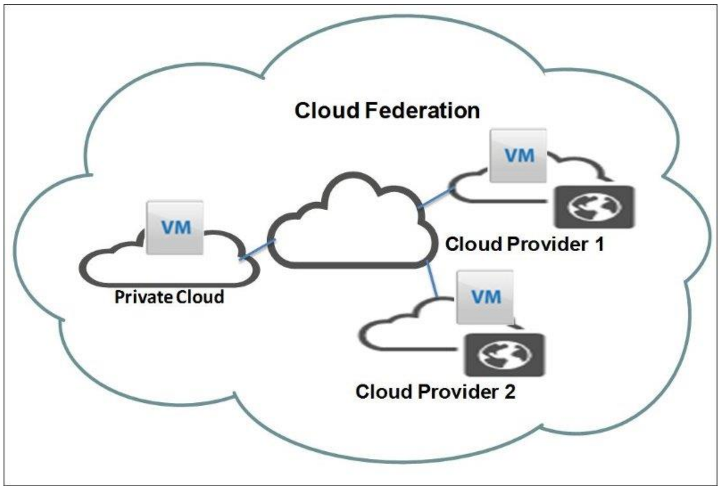


Figure 1: Cloud Federation

In a federated cloud environment, one CSP wholesales or rents out the computing resources to another CSP. These computing resources become available in the buying CSP for the duration as decided in a mutual agreement. By doing so, cloud federation offers following two advantages to the CSPs.

- It allows the home CSP to cater for the spikes in the demand of the customers while going unnoticed and also saves the investment of extending the resource pool. Scalability of resources also becomes simple and trouble-free.
- Allows foreign CSP to generate revenue from the underutilized / idle computing resources.

In the process of forming a federation, the participating CSPs establish a Service Level Agreement (SLA). The data/ services are then redirected from home CSP to Foreign CSP when a request from customer is received. this activity is seamless to the customer but it arises their concerns about privacy and security of data residing in a Foreign Cloud. To streamline this procedure, a need is felt for establishment of trust between the CSPs participating in a Federation. Excessive research has been carried out to establish and evaluate trust. Various trust models are in use, which calculate the trust value among the participating CSPs before they become part of a federation. The trust value itself is dependent on the performance and response of the Home or Foreign cloud as per the SLA agreed upon. Any deviation in the agreed upon parameters opens the way for mistrust just like in real life. Therefore, a need is felt to dynamically update the trust value as the behavior of participating CSPs varies which is the main focus of this research.

1.2 Research Significance

Evaluating and keeping a record of dynamically updated trust value is essential in order to have an accurate picture of the performance of CSP in a federation. The dynamically updated trust value would also save the trust calculation time, each time a new CSP wants to register in a federated environment. A real-time and readily available trust value would depict a true picture of a CSP. Existing research includes three to four SLA parameters for evaluation. This research would calculate trust value involving more SLA factors and hence a better trust value is concluded.

1.3 Relevance to National Needs

Since, majority of the IT, banking, commercial and business sector has moved onto cloud environment, on a national level this research can be used by all IT organizations whose operations are dependent on cloud environment or all those who are providing services as a CSP. Trust evaluation and continuous monitoring of the same, would help the organizations to assess the level of services provided by the CSP.

1.4 Motivation and Problem Statement

Cloud computing is a broadly accepted technology these days. The cloud customers are relieved from setting up and maintaining dedicated infrastructure. Customer's data and applications are deployed off site and are reachable to him independent of his location. All a customer needs is an internet connection to have an access to his resources. Despite the improved flexibility and ease of access to the users, the customers face security and privacy challenges that need be addressed.

There are certain circumstances where even a Cloud Service Provider does not possess sufficient resources to meet the changing requirements of the customer. The cost of deploying and extending the infrastructure at a given time might not be possible for the CSP. To cater for such situations, concept of Cloud Federation evolved wherein a CSP running out of resources looks for another that can collaborate and offer its idle resources under a service level agreement.

The trust of one CSP on another in a federation is proportional to the compliance of the SLA parameters. Numerous methodologies have been devised to calculate this trust value. However only a few exist that take into account the time factor and real time trust update. The aim of this thesis is to devise a trust mechanism that evaluates the SLA parameters in real time and dynamically updates the trust value on regular intervals. This would help the new CSP to rabidly shortlist the most suitable and trustworthy CSP to form a federation with. Figure 2 represents the problem statement of this research.

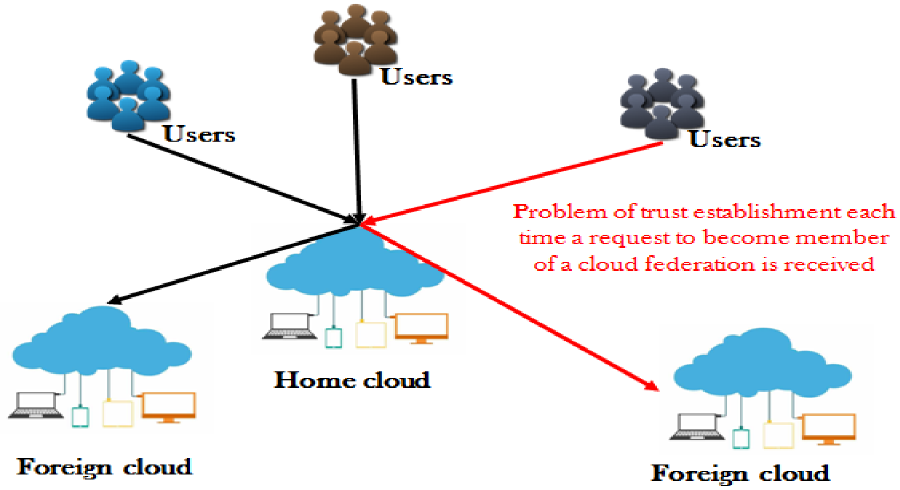

Users
Users
Users
Problem of trust establishment each time a request to become member of a cloud federation is received
Home cloud
Foreign cloud
Foreign cloud

Figure 2: Trust in Cloud Federation - Problem Statement

1.5 Contributions and Outcomes

Enlisted below are the contributions of this research work:-

1. Detailed study and analysis of the existing techniques for dynamically updating the trust value in federated cloud environment.

2. Propose an enhanced trust management model for updating trust value iteratively in cloud federation.

3. To evaluate the proposed trust model in order to validate its efficiency.

1.6 Thesis Layout

The presented thesis comprises of 5 chapters. It is structured as follows:

- **Chapter 2** discusses the history of cloud computing, its characterisctics. The chapter introduces the concept of cloud federation. CC service delivery models and deployment models along with the possible security challenges to the cloud computing technology. It is followed by a brief literature study of the existing techniqes discussed in terms of its weaknesses.
- **Chapter 3** presents the workflow, design and architecture with component lovel brief details of the proposed trust update model. The chapter also constitues the algorithm of the proposed model.
- **Chapter 4** discusses the implementation mechanism of the proposed Trust Update Model and the corresponding results.
- **Chapter 5** concludes the research work presented in this document and highlights open channels for future researchers in the field.

1.7 Conclusion

The chapter described the main objectives alongwith the motivation to carry out research. It also highlights the importance and relevance of research to the national needs. The chapter concludes with the structure of the thesis document.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

This chapter briefly outlines the history of cloud computing technology. Literature review and detailed study on cloud computing its characteristics and challenges is also listed. Evolution of the concept of Cloud Federation alongwith trust evaluation model studied have been shared.

2.2 History of Cloud Computing

The introduction to the concept of Cloud Computing dates back to 1960 when John McCarthy [3] proposed the idea of providing computing services as a public utility in the form of utility and grid computing. With the emerging use of internet in daily life, the concept enhanced into classification of IaaS, SaaS and PaaS [4] during 2007. Salesforce.com stood as first movers in the era of CC around the year 1999 followed by Amazon Web Service, Google , Microsoft Windows Azure and many more. The field is yet to be unveiled by the scientist for its magnificent utilization.

2.2 Characteristics of Cloud Computing

NIST definition of cloud computing [4] outlines its five essential characteristics. Each of them is briefly summarized below and represented in Figure 3:

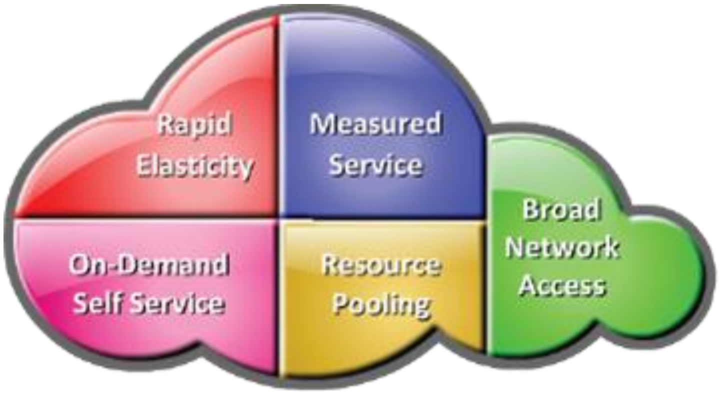


Figure 3: Characteristics of Cloud Computing

- **On-Demand Self Service**

  Customer provisions resources (compute, storage, processing etc) to itself without any requirement of human interaction.

- **Resource Pooling**

  Provider has an aggregate of resources (processing, memory, storage, compute, network bandwidth etc) to serve a wide range of consumers in a multi-tenancy model. Various physical and virtual resources are dynamically assigned to the customers in order to cope with their changing demand.

- **Broad Network Access**

  All cloud facilities are accessible and available to the customer for each platform (both thick and thin clients).

- **Measured Services**

  Resource utilization is automatically controlled and optimized by using some metering techniques to the services provided. Resource usage can be monitor, controlled and reported. This provides transparency to both cloud service provider and its consumer.

- **Rapid Elasticity**

  Cloud capabilities scale-in and scale-out with respect to the demand of the customer. Thought limited in actual, the resources seem to be infinite to the consumer.

2.3 Cloud Federation

Cloud federation is the aggregation of software, platform and infrastructure services from different CSPs that can be accessed by a consumer through internet. The concept offers following benefits [5][6]:

- Solves the problem of vendor lock-in.
- Optimizes enterprise IT service delivery.
- Allows a consumer to choose best CSP for its business and technical requirements.
- Consumer's application runs in most appropriate infrastructure environment
- Balances and distributes the workload among the trusted entities of the federation.
- The concept eliminates the possibility of single point of failure and secures the investment of consumer.
- Provides scalability and flexibility.

2.4 Cloud Computing Service Models

According to NIST, Cloud computing offers three service models according to the type f service delivered to the end-users. These models offer abstraction and are therefore often portrayed as layers in a stack. The three models are described below:

- **Infrastructure as a Service (IaaS)**

  This model makes infrastructure available to the customer over the internet by distributing the larger physical infrastructure into smaller virtual machines. Infrastructure composes of all kind of hardware, storage, servers, network and computing services. The customer saves his investment which he would otherwise spend in buying the hardware dedicatedly and the maintenance cost is also saved. The customer if free to install any software of his requirement on this infrastructure. The customer uses the infrastructure as long as required and can pull off his rights when required. In the conventional case, the infrastructure stays idle and unutilized.

- **Platform as a Service (PaaS)**

PaaS allows the customer organizations to build, deploy and manage the application without infrastructure. PaaS offers an environment to the user for development, testing and deployment without the need of worrying about the overhead of computing, storage and provisioning issues. All these facilities are scalable as per the requirement of the customer while doing the coding. This model is an equivalent to that of middleware in conventional computing.

- **Software as a Service (SaaS)**

SaaS offers cloud services at the application level on demand. The application is centrally hosted and made available to the users upon request. This bypasses the cost incurred in buying the licenses, its maintenance and installation charges. Microsoft and Google are the legendry examples of SaaS.

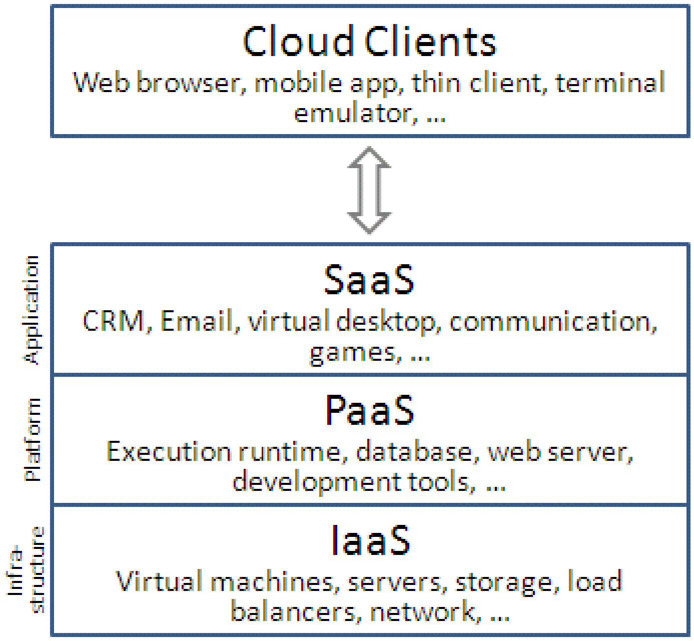

Cloud Clients
Web browser, mobile app, thin client, terminal emulator, ...
SaaS
CRM, Email, virtual desktop, communication, games, ...
PaaS
Execution runtime, database, web server, development tools, ...
IaaS
Virtual machines, servers, storage, load balancers, network, ...
Application
Platform
Infra-structure

Figure 4: Cloud Computing Service Models

2.5 Cloud Computing Deployment Models

Cloud computing technology is categorized with respect to their deployment strategies. The deployment model have been designed to meet the organization specific requirements. The four categories are defined below and depicted pictorially in Figure 5.

- **Private Cloud**

  The cloud infrastructure is exclusively available to a single organization. The infrastructure is owned and managed by the organization itself or a third party on-site or off-site. Private clouds are the capital intensive. Updating and maintaining the infrastructure incurs additional costs to the organization. Private cloud is best suitable for mission critical, and information sensitive organizations. It is the most secure and reliable service provisioning deployment model.

- **Public Cloud**

  The infrastructure and services are available for public use on sharing basis over the internet. The ownership, management and operations of the cloud are managed by a business, industrial or academic organization on-premises. The cost and expenditure are little less than that of private cloud because the services are more commoditized.

- **Community Cloud**

  Shared among more than one organizations that work as a community with a mutual business goal or mission to follow. Cloud is hosted either by one of the member organization and cost is distributed among few members or it may be externally hosted by a third party organization.

- **Hybrid Cloud**

Hybrid Cloud is a composition of two or more cloud deployment models (private, public, or community). Hybrid cloud is typically designed for cloud bursting or cloud peering. Hybrid Cloud model is ideal for load-balancing of the clouds.

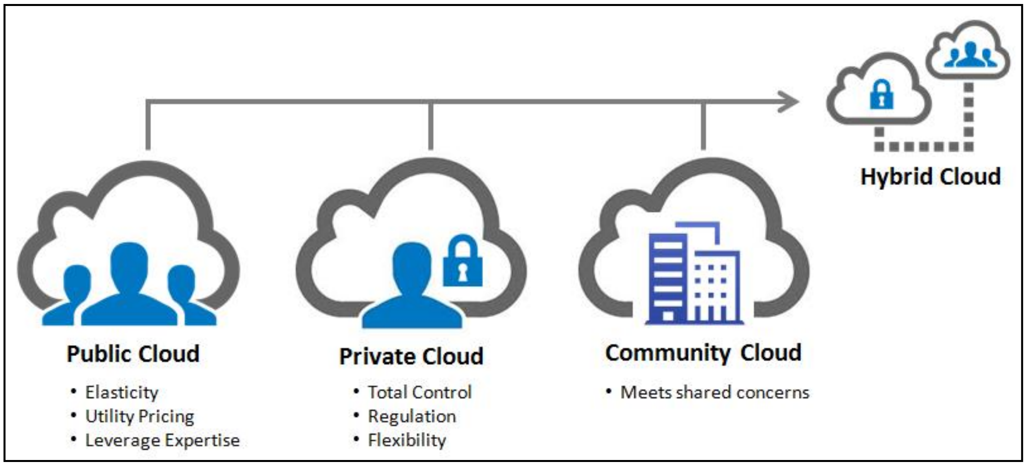


Figure 5: Cloud Computing Deployment Models

2.6 Challenges to Cloud Computing Technology

Cloud computing technology is emerging rapidly due to its wide deployment and broad acceptance across the globe. This calls for evaluation of security and privacy concerns by technologists. Users in general do not have the rights to monitor the granular details of the cloud architecture [7]. New attacks are being revealed in cloud computing architecture owing to its non-visibility of location of data, resources sharing over shared infrastructure and network, vendor lock-in etc. Other prime security challenging include data confidentiality, privacy and integrity alongwith outsourcing, multi-tenancy, trust establishment and identity management. Security challenges are briefly described below:

- **Data Security**

  Cloud computing technology is a reasonable option for all sorts of organizations. CSP meets the requirements of the user. CSPs need to implement enhanced security mechanisms to prevent major and minor data breaches and security attacks of their valuable customers. Strong cryptographic techniques, encryption schemes and access control mechanisms need to be implemented [8]

- **Location of Data**

Location independence is one of the prominent features of cloud computing technology. Data in a cloud environment resides at multiple location which are geographically apart from each other. The particular country's laws and regulation apply on the data that resides in their premises. This poses a serious concern to the customers who possess confidential data when organizations data crosses the boundaries for access or storage purpose [8].

- **Multi-Tenancy**

Data of one CSC might reside at different location. Similarly, data of multiple CSCs might reside in a shared location. Viruses and malicious codes might get transferred from one CSC to another being in same location. Proper control measures to segregate the data of all the CSCs in cloud computing environment are required like data validation mechanisms, SQL injection flaws,

- **Network Security**

Communication between CSPs and CSCs is carried out over a shared network. This demand high standard of network security measures to be implemented to avoid any attack on the network which could cause data loss to the consumer. Encryption techniques and secure communication protocols be implementation to ensure the same.

- **Access Management**

Due to multi-tenancy feature, multiple CSC's data reside in a common location. To ensure confidentiality of each CSC and protect is data, application and system, access to unauthorized person needs to be eliminated. Cross virtual Machine attack is one of such example which exploits the confidentiality of the system. This all requires the need for strict access control mechanisms to keep the unauthorized out while maintaining the availability to the right one.

- **Trust Management**

Trust is an important factor in cloud computing and federated cloud environment that ensures the confidentiality, integrity and availability of data to the consumers as and when required. Any deviation in the claimed features of a certain CSP reduces its trust among the CSC. Similarly in a cloud federation, the element of trust exists among the member CSPs of a federation as well as between the CSP and CSC. More, the trust value, more secure it is considered. Hence, trust management schemes are need of the time and technology. Existing trust mechanism models are discussed further in this chapter. From this point onwards, this document would revolve around trust management models being the focal point of this research.

2.7 Trust Evaluation Models - Overview and History

Trust is an essential ingredient to deliver reliable services to the consumers in a cloud computing environment. To ensure trust level among the CSCs, continuous compliance to the expected behavior of a cloud is necessary. owing to the dynamic nature of cloud computing environment , maintaining a certain level of trust in Cloud Computing environment is a challenging task. The concept of trust management was introduced by Blaze. M back in year 1996 [9].Gradually various trust models were proposed in the dimensions of ubiquitous computing, peer-to-peer networks, adhoc networks, multi-agent systems, wireless sensor networks, cloud computing and multi-cloud systems.

In a survey conducted by Hoffman and Novak in 1999, majority of the web users i.e. 95% showed mistrust in online shopping businesses on grounds of security and confidentiality of their personal information [10]. To cater for such issues and ensure secure and reliable online transactions over the web, a trust evaluation model was introduced in year 2000 by Machala [11].

This concept was further enhanced into the areas of wireless networks, peer-to-peer networks, grid computing and mobile networking etc. Paul Manuel proposed a QoS Trust in 2015. The QoS trust model evaluated trust factor based on past credentials i.e.

QoS provided by the CSP measured in terms of availability, reliability, data integrity and turnaround efficiency each with their own weightages [12]. Alhamad proposed the idea of calculating trust of a cloud based on SLA parameters as well as the opinion of external cloud providers [13]. The model lies under the category of SLA Based Trust Models. The trust management methodologies further moved into the category of interaction based architectures. In 2012, S.Ahmad and his co-authors [14] propped a model that evaluates the trust based on the interaction of CSP and CSC. The working of the model comprises of three stages which includes the gain of sufficient of knowledge of Cloud computing concept and a certain Cloud Service Provider. The second stage is passed when the user gets to know of all the merits and de-merits of using a particular cloud services. Based on these two stages, the CSC agrees to rely on the CSP after necessary satisfaction and starts using the services of the CSP.

Further additions to the trust management aspect include behavior based, attribute based, reputation based and feedback based trust computing models [15][16][17]. The focus of this research is onto the trust establishment between two or more cloud service providers that are part of a federation to meet the requirements of the CSCs.

2.8 Trust in Cloud Federation - Existing Schemes

Some notable trust models for a federated cloud environment after detailed literature study are enlisted below with analysis.

Table 1: Literature Review of Existing Techniques For Trust Establishment and Evaluation

Year	Author(s)	Paper	Description	Weakness
2017	Zhenhua Tan, Yicong Niu, Yuan Liu, Guangming Yang [18]	A Novel Trust Model Based on SLA and Behavior Evaluation for Clouds	Proposed a model based on SLA, uses fuzzy logic to evaluate the compliance of SLA parameters in transaction history	Limited SLA parameters considered.
2017	Zhen-Hua Tan, Yi-Cong Liu, Nan-Xiang Shi, Xing-Wei [19]	MCTModel: A Multi-clouds Trust Model	Proposed method updates trust value based on	Limited SLA parameters considered.

		Based on SLA in Cloud Computing	CSP’s SLA performance against the service usage.	
2016	Matin Chiregi, Nima Jafari Navimipour [20]	A new method for trust and reputation evaluation in the cloud environments using the recommendations of opinion leaders' entities and removing the effect of troll entities	Proposes a method to evaluate trust based on QoS parameters and identifies troll entities in a federation.	Lacks trust update module.
2016	Zhenhua Tan,XingweiWang,and Xueyi Wang [21]	A Novel Iterative and Dynamic Trust Computing Model for Large Scaled P2P Networks	Model presents an iterative and dynamic trust computation method based on historic transactions and feedback collected in an iteration.	Limited to P2P systems
2014	Ayesha Kanwal, Rahat Masood, Awais Shibli [22]	Evaluation and Establishment of Trust in Cloud Federation	Trust of CSP in a federation is calculated based on feedback of users and QoP parameters extracted from SLAs.	Feedback of a user is subjective and static values of SLA QoP parameters are being used.
2013	Xiaonian Wua, Runlian Zhanga, Bing Zengb, Shengyuan Zhou [23]	A Trust Evaluation Model for Cloud Computing	Proposed system which uses probability theory on recent interactions and ratings of entities (CSPs) for estimating and updating trust.	Not based on SLA parameters.
2011	Mohammed Alhamad [24]	SLA-Based Trust Model for Secure Cloud Computing	Proposed model is based on fuzzy inference system and evaluates the parameters on	▪ Limited SLA parameters being used.

			reputation of a CS and its recommendations by a 3rd party agent.	▪ Reputation is a relative term.

Table 1 outlines the salient of literature review of existing mechanisms to establish and evaluate trust of a CSP in CC environment. It is obvious from the above enlisted summary that each techniques possesses strengths as well as weakness. Majority of the mechanisms revolve around a static and one-time calculated value of trust at the time a CSP wants to join federation. Therefore, a need is felt for a mechanism to evaluate trust value dynamically and make it readily available when required.

In [18], authors presented a trust model based n behavioral factors alongwith SLA parameters. Parameter vectors are continuously formed during the transaction history which contribute in dynamic change of trust value based on behavior of the CSP. The simulation results have proved this technique to be effective.

In [19], authors proposed a mechanism to evaluate and update trust using time-decay model in a Multi-Cloud environment. The consumer agrees on type and level of service expected from a CSP during service usage in the form of SLA. These expected values are monitored as per the weightages mentioned by the consumer against availability, reliability and integrity parameters. The model is focused on service provisioning to single consumer in a multi-cloud environment.

In [20], trust value is evaluated on five parameters: availability, reliability, integrity, capability and identity alongwith the identification of troll entities in CC environment based on opinion leaders recommendation.

In [21], authors have presented a simulation result and analysis of trust calculated based on historic data of transactions carried out in a Peer-to-Peer network communication. The approach is limited to P2P systems and possesses least relevance to the cloud federation environment.

In [22], authors have proposed a protocol to evaluate trust in Federated Cloud environment based on combined value of parameters extracted from SLA and feedback regarding the security and privacy of data provided by the CSP. Value of trust is calculated when a request for federation is received and remains static. The values of SLA parameters are also extracted from the claims of CSPs in SLA rather than real time service utilization statistics.

In [23], the proposed model is based on results of feedback of mutual interaction among the CSC and CSP. Both the CSP and CSC evaluate eachother over a service provided. Sliding time window concept has been used to ensure the efficacy of transactions over a period of time. This gives a realistic picture of how transactions are being carried out in actual and its effect is reflected in corresponding calculations. The model also involves element of reputation which a subjective evaluation of entities. Trust value calculations have been performed by D-S evidence theory concepts.

In [24], *Mohammed Alhamad* presented a trust model that includes standardized criteria for the service level agreements to evaluate the level of trust upon cloud providers. The proposed model is applicable on all types of computing environments i.e Grid Computing, P2P Networks and Cloud Computing etc. The trust value calculation also involves reputation element.

2.9 Conclusion

This chapter briefly summarized history, characteristics, service and deployment models and various security and privacy challenges in the CC environment. History of various trust evaluation models in cloud computing have been discussed followed by a brief analysis of trust establishment and evaluation models proposed by authors across the globe in Federated Cloud environment. In the next chapter, we explain our proposed trust evaluation and update model

Chapter 3

PROPOSED MODEL FOR UPDATING TRUST VALUE IN CLOUD FEDERATION USING ITERATIVE APPROACH

3.1 Introduction

This chapter presents the crux of the research. It revolves around the concept and workflow of the proposed model in a federated cloud environment to evaluate and update trust. The design and architecture along with a brief description of the composing modules of the proposed model is discussed.

3.2 Proposed Model for Updating Trust Value In Cloud Federation Using Iterative Approach

Techniques and models exits to evaluate and establish trust value of a certain CSP. Majority of which calculate the trust value once when a CSP wants to register in a federation. The trust value established is calculated with a limited number of SLA parameters. We propose a trust model which add on to the number of SLA parameters in consideration to have a better view of reliability of service provisioning on a CSP. Also an update module is introduced to ensure that trust value is readily available whenever a new CSP wants to register which is a true reflection of the CSP. The trust value is updated in an iterative manner. The proposed model is comprised of following modules:

- CSP Registration Module
- SLA Management Module
- SLA Monitoring Module
- Trust Evaluation Module
- Trust Update Module

The trust factor thus calculated takes on an arithmetic value ranging between 0 and 1 with 0 being the minimum and 1 being the maximum. Architectural overview of the proposed model is presented below followed by description of composing modules.

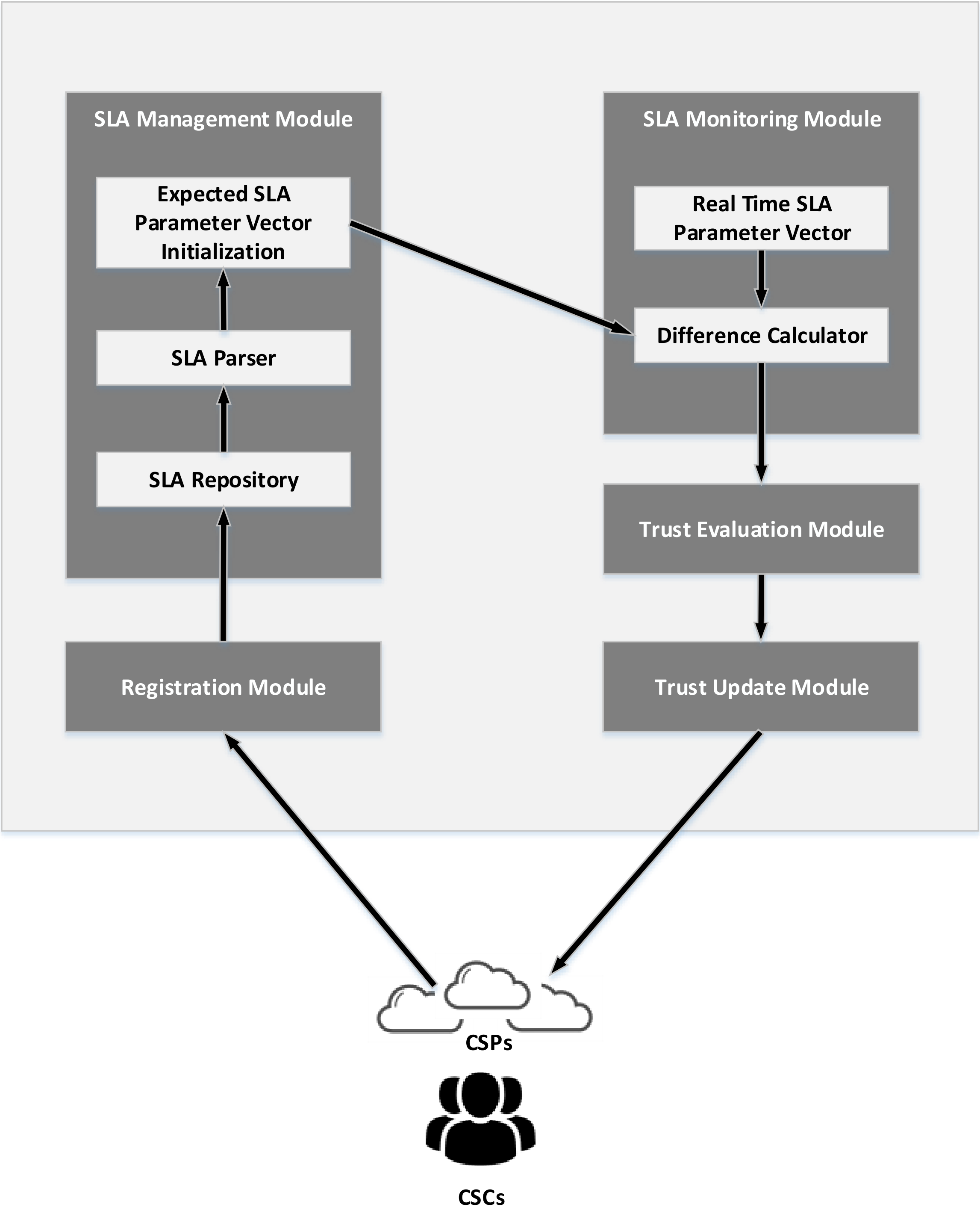


Figure 6: Architecture of Proposed Model

As a very first step, the CSP wishing to become part of a federation , gets itself registered via registration module after necessary authorization. During the registration, the CSP submits information of its primary customers, general information, SLA and the services it provides to the customers.

SLA collected by the CSP are then processed through the SLA Parser to extract the parameter-value vector which is hereby called as the expected SLA parameter vector and shall be used for comparison in upcoming stages.

The presented model revolves around comparison of services promised by a CSP in their SLA vs. the services it is offering in real time environment. The comparison is carried out by the Difference Calculator in SLA Monitoring Module based on the transaction history and performance values of the CSP. The SLA parameters in consideration are compared to see for any deviation and passed onto the Trust Evaluation Module to calculate its value.

For every comparison entry that takes place, trust value is updated based on degree of fulfillment of SLA parameters. This iterative calculation process also takes into account the time factor to ensure that a CSP is penalized in terms of trust value if it shows up a degraded performance.

The updated trust value that resides now for a CSP is the readily available trust factor that helps other CSPs while joining a federation. Work flow diagram of the presented model is shown below in Figure 7:

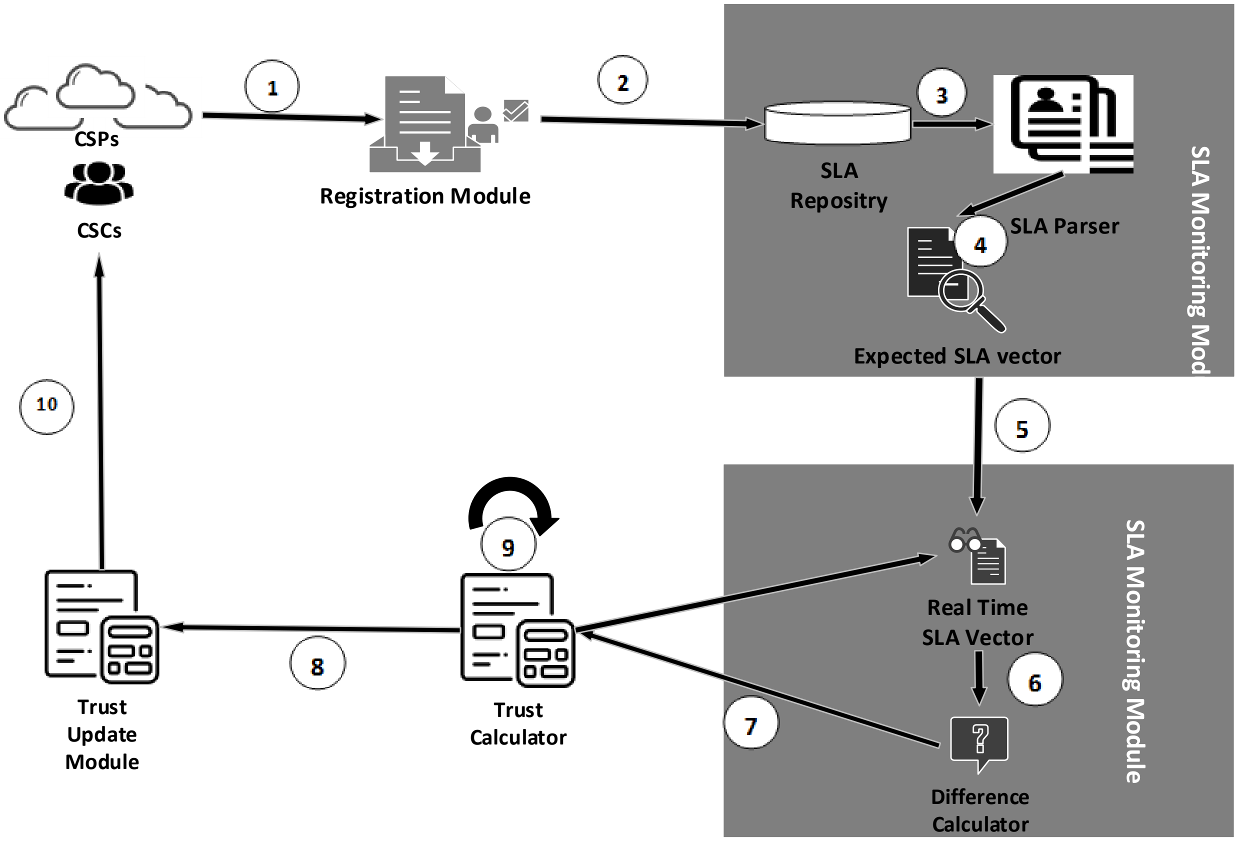


Figure 7: Work Flow Diagram of Proposed Model

3.3 Components of Proposed Model

The architecture of proposed model has been presented above in Figure 6. The main components of the model comprise of Registration Module, SLA Monitoring Module, SLA Management Module, Trust Evaluation Module and rust Update Module. Each of these components is explained below:

3.3.1 Registration Module

The registration module is the entry point of a CSP into a federated environment. This module is responsible to register the users and CSPs who wish to join a Cloud Federation. During registration, documents are collected and information regarding the services offered by a CSP is collected and stored centrally. Among all of which, the SLA document holds prime importance. This documents is forwarded to SLA Management Module for further processing.

3.3.2 SLA Management Module

Service Level Agreement (SLA) is a highly significant document that outlines the agreed upon services, parameter value set of QoS parameters promised between the two parties and compensation factor or procedures in case of violation to come upto the expected level. SLA is brief enough to include details from definition of services to termination of agreement, including the specific terms and agreements about the rewards as well as penalties. SLA also mentions the period after which it needs to be revised. In short, this contract determines the quantifiable value of the services that a Cloud Service Provide shall provide and outlines the compensation to the consumer in case of failure. Figure 8 depicts the main concepts which form part of a well-defined SLA document [25]:

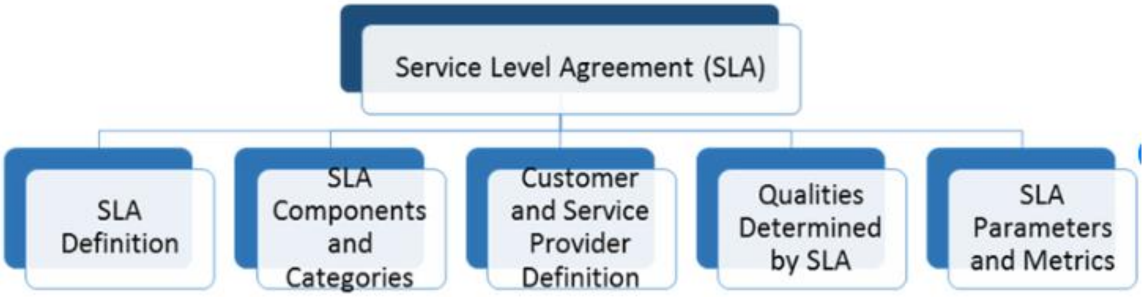

Service Level Agreement (SLA)
SLA Definition
SLA Components and Categories
Customer and Service Provider Definition
Qualities Determined by SLA
SLA Parameters and Metrics

Figure 8: Main Concepts of SLA

A well defined SLA includes following specifications:

- The type of service provisioning.
- The desired level of performance in terms of QoS of each service.
- Monitoring process and service level feedback/ reports by gathering requisite statistics.

- Procedure to report a certain anomaly in the services alongwith the contact information of concerned administrator.
- Response time-frame and Issue resolution time.
- Penalties to be imposed on CSP in case of failure to provide agreed upon services.
- Billing method for the agreed upon services.

A well-structured SLA is equally beneficial to both the parties signing the agreement. Following are a few advantages of a briefly defined SLA.

- **Improved Customer Acceptance Level**: A rightly explained SLA ensures increase in the confidence of consumer in a CSP and its services. Compliance to SLA agreed QoS parameters enhances the trust of consumer and encourages more users.
- **Enhanced Service Quality**: SLA includes Key Performance indicators (KPIs), which tracks the customer service provided by CSP and tracks the compliance of outlined indicators to the requirements of CSPs and CSCs. This check on services of a CSP leads to enhanced quality of service.
- **Strong Relationship Between the Signing Parties**: SLA brings forth the quantitative value of the expected services and deliverables. Payment and payback procedures are clearly shared between the two parties. SLA also states way out to resolve any contractual disagreements.
- **Fault Tolerance**: To cater for delayed response and latency issues, the CSP devises mechanism to catch and resist faults and crashes. This results in fault tolerance if a CSP.

An idyllic SLA is composed of components defined below [25]:

- **Purpose**: State the aim of formulating SLA.
- **Parties**: Declares the stakeholders of the SLA and their job.
- **Scope**: Describe the services mentioned in the SLA for easy recognition of the SLA services by the consumer.
- **Validity Period**: Duration (start and final time period) for which the terms and conditions of the SLA are applicable and valid.
- **Restrictions**: Procedure and essential activities to be performed to provide required level of service.
- **Service Level Objectives**: A group of service level indicators comprising of QoS parameters such as availability, reliability etc alongwith their validity in day-time frame and target level to achieve.
- **Service Level Indicators**: Declare the indicators to use to gauge the quality of service.
- **Penalties**: Action to be taken in case of non-compliance to the promised services in SLA. Declare the compensation to be provided to the customer and if stated the procedure to conclude the agreement.
- **Administration**: Define procedures formed to attain and evaluate SLA.
- **Optional Services**: Describe the services that are not essentially required to the consumers but might be required as exclusion.
- **Exclusions**: Specify what all is not part of SLA.

Figure 9 represents the composition of a clearly defined SLA.

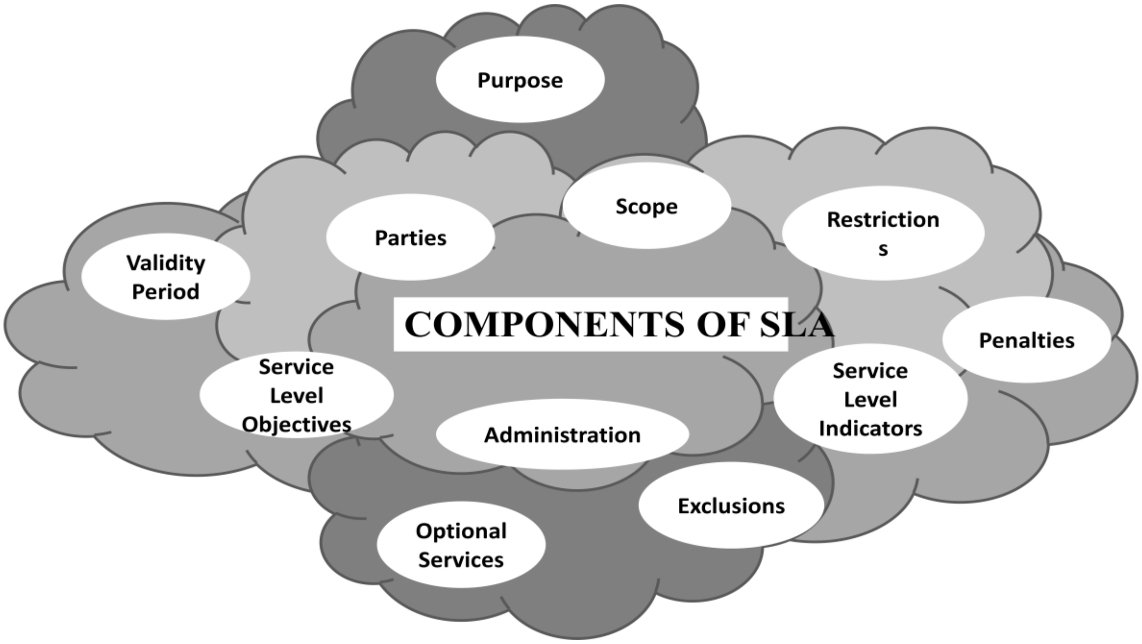


Figure 9: Components of SLA

SLA documents lack a standard format. Each agreement creates a format of its own to follow. However, generally the parameters are common. Below, we will outline generic and most considered metrics mentioned in SLA as per the deployment model of the Cloud [25]. Metrics form the basis of SLA parameters for QoS monitoring. These metrics help in accountability of services by consumer.

o SLA Parameters for IaaS

IaaS deployment model provides virtualized computing resources to the consumers alongwith the hardware platform. Significant SLA parameters in IaaS deployment model are stated below:

Parameter	Description
CPU Capacity	The processing speed of a VM .
Memory	Cache memory of VM
Boot Time	Time taken to boot and get ready for provisoning of services.
Storage	Storage held with the IaaS cloud for the validity

	period of SLA.
Scale Up	Maximum amount of VMs per user.
Scale Down	Minimum amount of VMs per user.
Scale Up Time	Time take to scale up (increase the number of virtual machines).
Scale Down Time	Time taken to scale down (decrease the number of virtual machines).
Auto Scaling	Option to automatically scale up/ down.
Availability	Duration for which services are available to the user.
Response Time	Time taken to complete assigned task.

Table 1: SLA Parameters for IaaS

SLA Parameters for PaaS

PaaS model allows its customers to develop and manage its own applications while the underlying platform is provided by the CSP. Standard SLA parameters being monitored in PaaS model are mentioned below:

Parameters	**Description**
Integration	Compatibility with other e-services and platforms.
Scalability	Degree of use with increase in number of users.
Pay as per use Billing	Payment as per utilization of resources.
Deployment Environment	Offline support.
Browsers	Compatibility with all available explorers.
Number of Developers	Number of developers who have access to the platform.

Table 2: SLA Parameters for PaaS

- ### SLA Parameters for SaaS

SaaS provides licensed software subscription to the customers while the software is centrally hosted. Common SLA parameters for this model are described below:

Parameters	Description
Availability	Time for which the software is available to the users.
Reliability	Ability to keep operating.
Scalability	Ability to scale up/ down the users of software.
Usability	Ease of use by the customers in terms of user interface.
Customizability	Flexibility of use by all types of users

Table 3: SLA Parameters for SaaS

Components of SLA Management Module:

Figure 10 below represents the components of proposed SLA management module followed by detail of each sub-module.

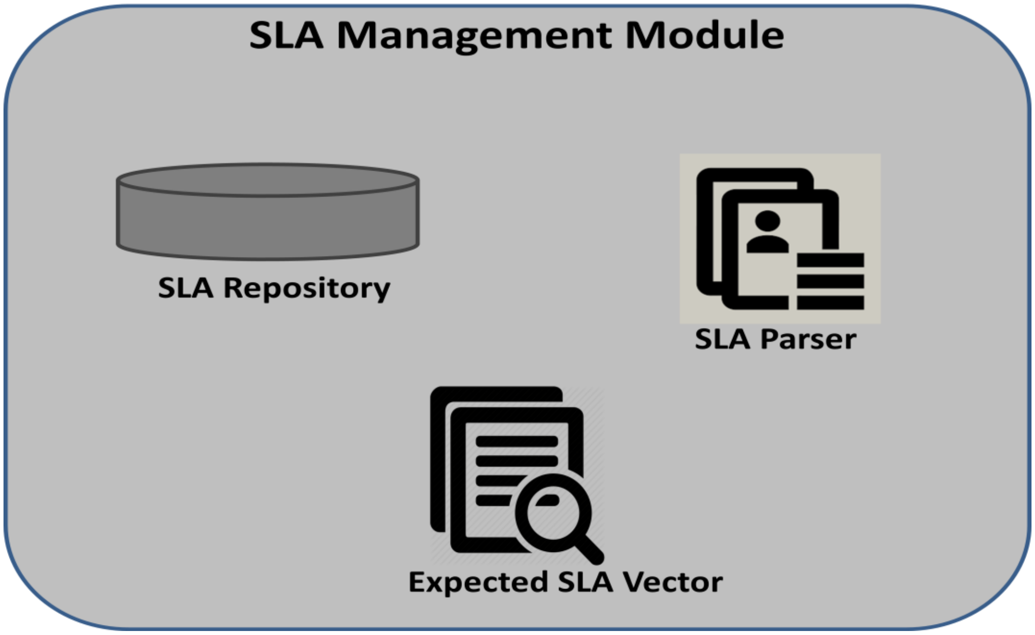


Figure 10: Components of SLA Management Module

a) SLA Repository

SLA document of the registered CSPs are stored in this repository. Unique identification number, name and URL pair is used to differentiate each CSP. This document is further passed onto SLA parser for processing.

b) SLA Parser

As the name dictates, SLA document received from SLA repository is processed for extraction of SLA parameters and corresponding value. The parameters discussed above coupled with a few others as mentioned in the individual SLA document are extracted for QoS monitoring and evaluation of trust value.

c) Expected SLA Vector

SLA parameter values as promised by the CSP forms Expected SLA vector. SLA parameters under study in this research comprise of Availability, Reliability, Integrity, Uptime and CPU capacity. Data of Expected Parameter Vector is represented by column vector as fol:

$$V_e = \begin{bmatrix} Av \\ Re \\ Int \\ Up \\ CPU \end{bmatrix}$$

Each of the SLA parameter is assigned a weightage, depending upon the preferences of customers of the CSP.

3.3.3 SLA Monitoring Module

Components of SLA Monitoring Module

Figure 11 below represents the components of proposed SLA monitoring module followed by detail of each sub-module.

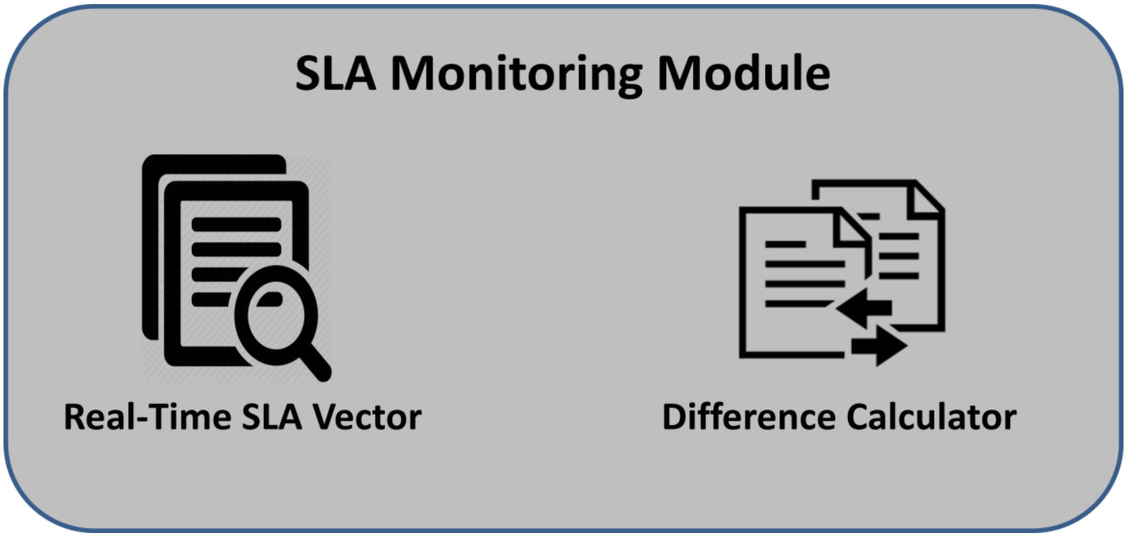


Figure 11: Components of SLA Monitoring Module

a) **Real-Time SLA Vector**

This module gets value of the SLA parameters under consideration as of real performance exhibited by the CSP. The CSP is continuously monitored and Real time SLA vector is updated on periodic basic. This vector is further used in evaluation of trust value. It is also a column vector represented by $\boldsymbol{V}_r$.

b) **Difference Calculator**

This sub-module is responsible to calculate the deviation of SLA performance as depicted in real-time by CSP from that of the promised value as dictated in the SLA document. Difference is calculated on every iteration after a periodic time. Difference Calculator module checks for compliance and fulfillment of service parameters. Fulfillment of SLA parameters retains/ increases the trust factor. On the other hand, non-compliance to SLA document results in a decrease in the trust value and also calls for a penalty. Penalty is imposed as mentioned in the SLA document. Similarly, a non-compliance results in penalty in the form of trust value. Trust value is decreased by a certain value (which is discussed later in this document).

3.3.4 Trust Calculation Module

This module is the heart of proposed model. its responsibility is to evaluate a quantitative value of SLA based trust depending upon the SLA parameters under consideration. The trust value of each iteration in a periodic span of time is calculated based on the compliance of parameters to the degree of service promised by the CSP. If the services delivered comply to the promised level, the trust value retains or increase up by a certain factor. In the other case, the trust value is declined and a penalty factor is imposed to the CSP as decided by the SLA document.

3.3.5 Trust Update Module

This module is repository of trust value exhibited by the CSP at each iteration. The trust factor evaluated periodically is saved by this module ad same is shared with CSPs and CSCs for their decision making. The trust value retains for the period of time till the next iteration is performed. After each interval of, the trust value is refreshed from the latest value calculated by the Trust evaluation module.

3.4 Proposed Algorithm to Evaluate Trust

The algorithm is composed of weighted values assigned to each SLA parameter as

per the priority of CSC. Weights are assigned to each parameter such as the sum of all of them equals 1. The corresponding set of weighted values is represented as a row matrix and represented as ***W***.

$$W= [w1 \quad w2 \quad w3 \quad w4 \quad w5]$$

such that

$$w1 + w2 + w3 + w4 + w5 = 1$$

And the weighted valued are denoted as a row matrix and denoted as WSLA parameters under consideration in this research form up the SLA vector comprising of five SLA parameters i.e. Availability, Reliability, Integrity, Uptime and CPU usage. The vector is denoted as column matrix, $\boldsymbol{V_e}$ represents the promised SLA values in the range of 0 and 1 whereas $\boldsymbol{V_r}$ represents the real-time values of these parameters exhibited by the CSP during service provisioning. The SLA vector is generally denoted as follows:

$$V = \begin{bmatrix} Av \\ Re \\ Int \\ Up \\ CPU \end{bmatrix}$$

where $\boldsymbol{Av}$ represents the Availability, $\boldsymbol{Re}$ represents reliability, $\boldsymbol{Int}$ represents integrity, $\boldsymbol{Up}$ represents Uptime and $\boldsymbol{CPU}$ represents CPU usage of the CSP.

- **Availability**. Availability refers to the access of cloud resources/ services to its authorized customers as and when requested. It is measured in terms of average response time.

- **Reliability**. Reliability refers to success rate of the tasks assigned to CSP. Success rate is the ratio of tasks completed successfully by CSP to total number of tasks submitted to a CSP.
- **Integrity**. Integrity is the assurance that data residing with a CSP is uncorrupted, unmodified and lossless.
- **Uptime**. It refers to the availability of CSP. It measured in terms of ICMP echo response of a CSP.
- **CPU Utilization**. This parameter reflect the CPU consumption of a CSP. CPU utilization is the workload being handles by a set of machines under the CSPs. The more a value of CPU utilization, the better is its performance.

The SLA vector $\boldsymbol{V_e}$ (expected SLA parameter vector) remains static for a given CSP, but the $\boldsymbol{V_r}$ (real-time SLA parameter) vary with reference to the performance of a CSP.

- **Deviation of Fulfillment**. The trust evaluation process highly depends on the deviation of SLA parameters. The fulfillment of promised SLA parameters is observed after measuring the change between the promised values and the exact values shown by the CSP in run-time with respect to their corresponding weightages. If the change is positive, it reflects that the services are up to the mark and as required. On the contrary, an unsatisfactory service level is exhibited by the CSP. This deviation is denoted as $\boldsymbol{DF}$ and calculated as below:

$$\boldsymbol{DF = W * (V_e - V_r)}$$

The above stated expression produces a scalar value which might be negative or positive as discussed earlier. The positive output represents that the fulfillment of service is higher than that promised by a CS. However, a negative value reflects that the fulfillment of service is lower than the promised level of service and doesn't meets the requirement of the customers. The degree of decay of fulfillment is denoted as $\boldsymbol{d}$ and represented by the equation stated below:

$$d = \begin{cases} 1 & ; \quad if\ DF > 0 \\ \gamma^{DF} & ; \quad if\ DF < 0 \end{cases}$$

- **Iterative Trust Update**. The trust value is updated in an iterative manner according to the fulfillment of service. If the value of $\boldsymbol{d}$ exceeds a certain threshold $\boldsymbol{T}$, an anomoly is recorded, the transaction is marked as a failed transaction and SLA based trust value $\boldsymbol{sT}$ is declined as dictated by expression below:

$$sT_t = \begin{cases} 0 & ; \quad if\ DF < 0 \quad and \quad sT < 0 \\ sT_{t-1} - \frac{1-d*sT_{t-1}}{\varphi} & ; \quad if\ \ DF < 0 \quad and \ \ sT > 0 \end{cases}$$

where $\boldsymbol{t}$ denotes current time and $\boldsymbol{t\text{-}1}$ is the previous transaction. The trust value can't be negative and hence its minimum value is set to 0. On the other hand, if the value of d is within the limits, the normal service provisioning is indicated and the CSP is encouraged in the form of increase in trust value as represented by equation below:

$$sT_t = sT_{t-1} + \frac{1-d*sT_{t-1}}{\mu} \qquad ; if\ DF > 0$$

where $\boldsymbol{\mu}$ represents reward factor and $\boldsymbol{\varphi}$ represents penalty factor and $\boldsymbol{\mu} \geq \boldsymbol{\varphi}$ because the intensity of punishment must be greater than that of reward.

- **Time Factor**. Cloud environment is dynamic in nature and trust value is greatly affected by the presence. absence of transactions being carried out. In the absence of a transaction, i.e. when a CSP is idle, trust will decline. The decay factor is calculated as an exponential function as defined below:

$$\boldsymbol{\rho = e^{-\beta * t}}$$

$$\boldsymbol{t = t_{current} - t_{previous}}$$

where $\boldsymbol{\beta}$ is decay rate, **t** is the time difference and $\boldsymbol{\rho}$ denotes decay factor. The closer the transactions, the more reliable a CSP is.

- **Pseudo code of Trust Evaluation Algorithm**.

The complete procedure to calculate the trust value is summarized below in the form of pseudocode:

Algorithm: calculateTrust(w1, w2, w3, w4, w5, Av, Re, Int, Up, CPU, Output sT)

```
begin
$t = t_{current} - t_{previous}$
$W = [w1 \quad w2 \quad w3 \quad w4 \quad w5]$
$DF = W * ( V_e - V_r )$
if (DF > 0) then
        d=1
        $sT_t = sT_{t-1} + \frac{1 - d * sT_{t-1}}{\mu}$
else
        $d = \gamma^{DF}$
        $sT_t = sT_{t-1} - \frac{1 - d * sT_{t-1}}{\varphi}$
```

This calculateTrust method is called for each iteration at a periodic interval and the evaluated value sT is stored as the latest one for the CSP under consideration. Parameter initial values are enlisted below:

Parameter	Notation	Initial Value
SLA Based Trust	sT	0.5
Time decay rate	β	0.005
Fulfillment decay factor	γ	2
Reward factor	µ	10
Penalty Factor	φ	5

Table 4: Parameter Initial Value

3.5 Conclusion

In this chapter, trust update model using iterative approach has been presented alongwith the algorithm/ pseudocode taking SLA parameters into consideration for evaluation purpose. In the chapter to follow, results of practical implementation shall be presented and analysis is carried out over the existing and proposed technique.

Chapter 4

IMPLEMENTATION AND ANALYSIS

4.1 Introduction

This chapter presents the details of proposed Trust Update Model for Cloud Federation using Iterative Approach. The proposed model is composed of five Modules as discussed in previous chapter. SLA parameters are taken as input. The values of SLA parameters used for reference purpose are extracted from SLA document of the CSP by parsing the document. Compliance to these parameters is checked at run time using monitoring tools as discussed in this chapter.

4.2 Experimental Environment

The proposed Trust Update Model is implemented in Visual Studio - C#. Cloud environment is highly empowered by the concept of virtualization. VMware offers a range of software to provide virtualization solutions. Cloud environment is setup in vCenter v6.5 with vSphere WebClient on client end for monitoring of the cloud. Physical blade servers mounted in E6000 chassis have been used, ESXi has been installed to virtualize these physical servers. Virtual machines have been created using VMware and vCenter is responsible for central server end management of these VMs and ESXi hosts.

For the client/ consumer end, vSphere Cleint has been installed on a Windows machine. vSphere client connects to ESXi servers and does the management and monitoring tasks. General component diagram of virtualized environment is as fol:

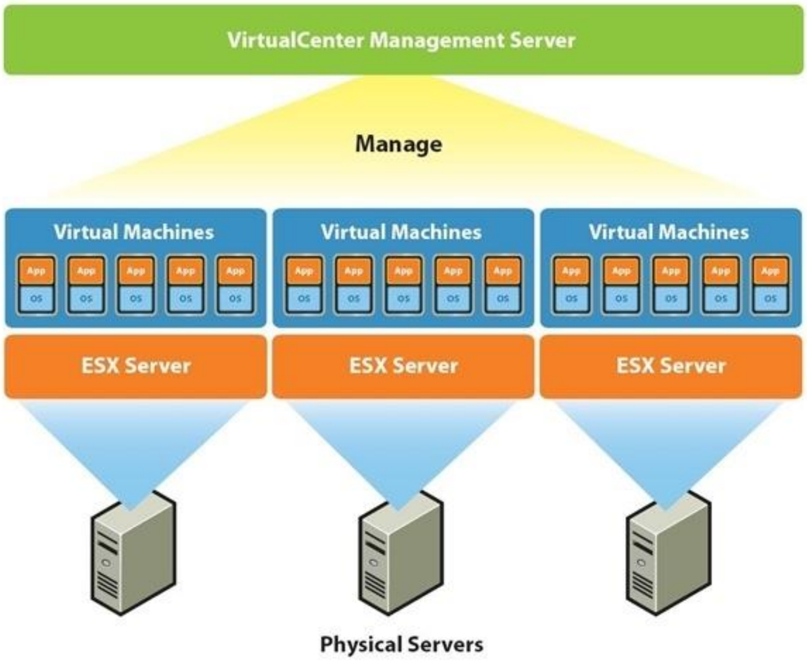


Figure 12: Components of Virtualization

4.3 Implementation of Proposed SLA Based Trust Update Model Using Iterative Approach

The process starts when the two CSPs want to be a part of federation to share their available resources, compute power, storage etc and get the maximum benefit out of the underutilized resources of each. The quantitative values of the SLA parameters under study i.e. Availability, Reliability, Integrity, Uptime and CPU utilization are gathered from the Monitoring and Performance options of vSphere Web Client [26]. The results are generated in percentages and exported in .csv format. For utilization in algorithm of the proposed trust model, the numeric values of each parameter are scaled to [0,1] range.

The history of performance is collected with a time interval of two hours. The trust algorithm takes these five SLA parameter values and their corresponding weightages as input from the csv file, runs over the algorithm and saves the calculated trust in the same csv file. Consequently, trust of the CSP is calculated and updated after every two hours. SLA parameter values for 100 periodic time intervals (two hours each) scaled to [0,1] range is shown in Figure 13 below:

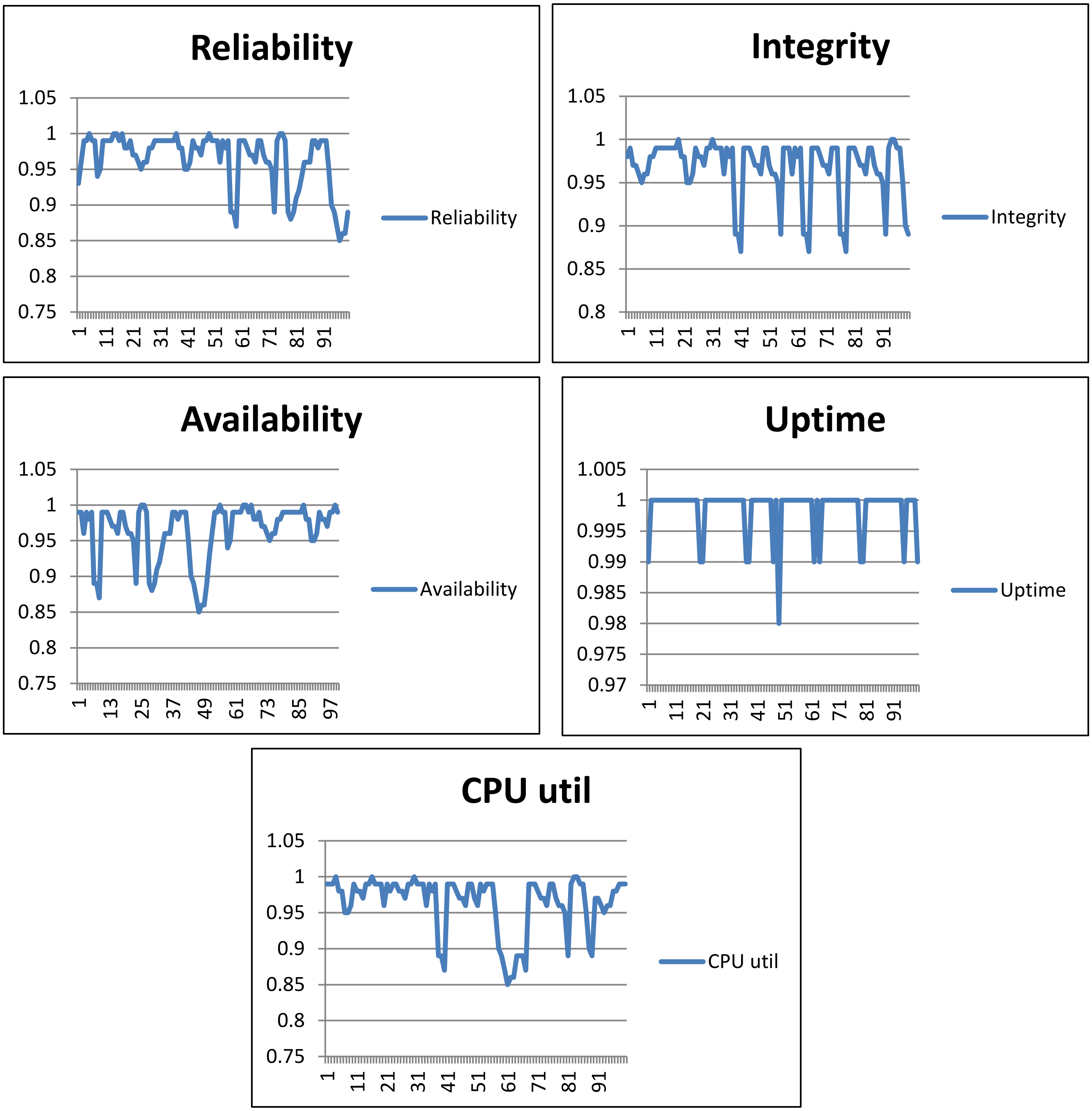


Figure 13: SLA Parameters Performance Chart

All of the above 5 SLA parameter set was taken as input with corresponding weightages as W= [0.3 0.1 0.2 0.2 0.2] and the corresponding value of trust is graphed as below in Figure 14 for all the 100 periodic time intervals:

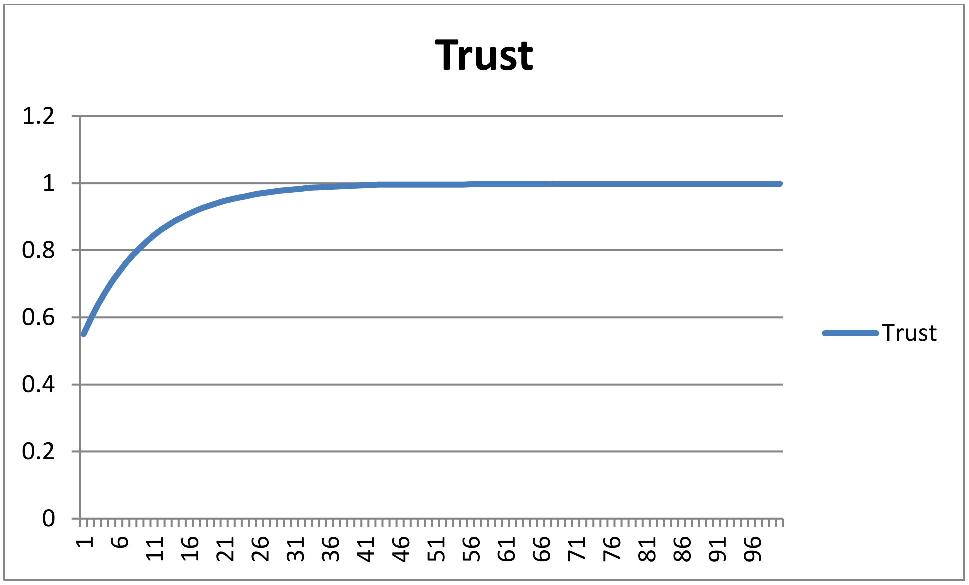
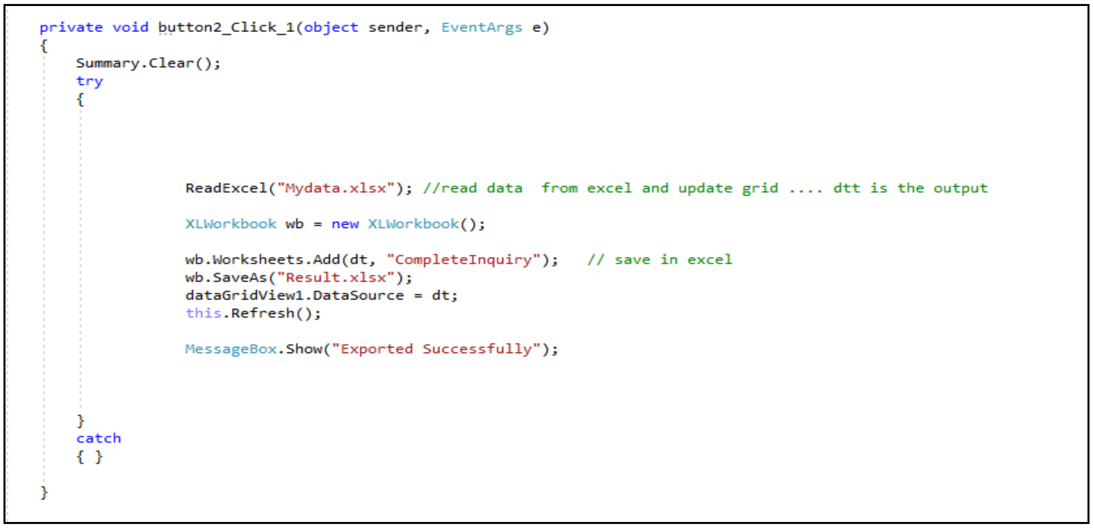


Figure 14: Trust value of CSP

From above presented graphical view of Trust value, it is clearly depicted that the trust value starting at an intermediate value of 0.5 gradually increases to maximum value of 1 and remains stable. This represents a CSP which is performing exceptionally up to the mark and as per performance offered to the customer in its respective SLA.

4.4 Implementation in C#

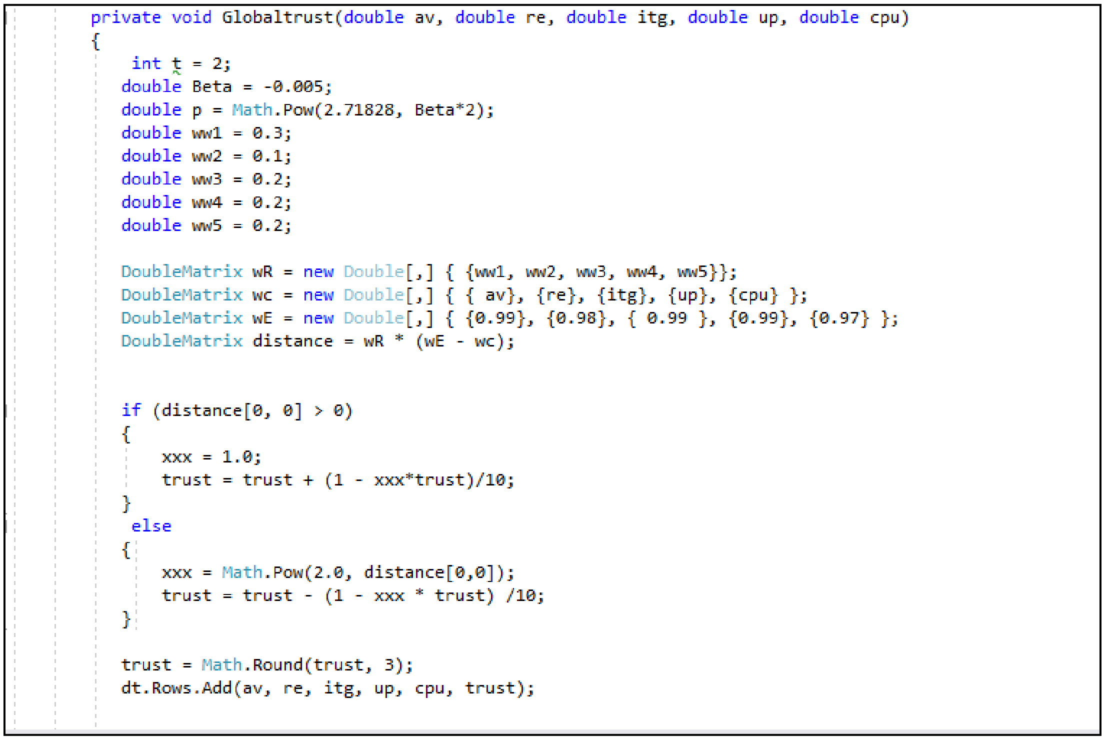
Performance monitoring values exported to an excel sheet and scaled in the range [0,1] will be accessed through a excel workbook (csv format) variable:



```
private void button2_Click_1(object sender, EventArgs e)
{
    Summary.Clear();
    try
    {



            ReadExcel("Mydata.xlsx"); //read data  from excel and update grid .... dtt is the output

            XLWorkbook wb = new XLWorkbook();

            wb.Worksheets.Add(dt, "CompleteInquiry");   // save in excel
            wb.SaveAs("Result.xlsx");
            dataGridView1.DataSource = dt;
            this.Refresh();

            MessageBox.Show("Exported Successfully");



    }
    catch
    { }

}
```

Figure 15: Read SLA parameter values

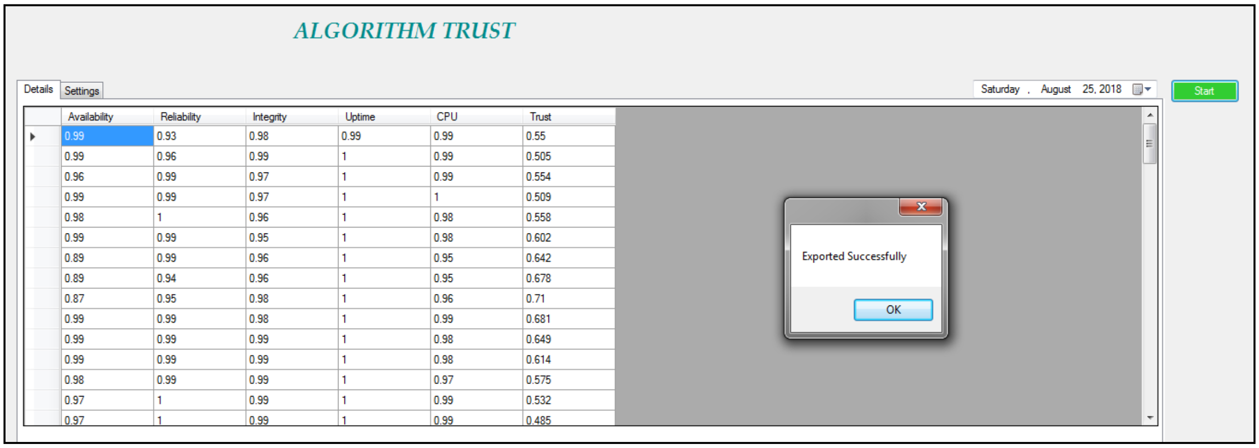
These values are passed onto GlobalTrust function for evaluation of trust value:



```
private void Globaltrust(double av, double re, double itg, double up, double cpu)
{
     int t = 2;
    double Beta = -0.005;
    double p = Math.Pow(2.71828, Beta*2);
    double ww1 = 0.3;
    double ww2 = 0.1;
    double ww3 = 0.2;
    double ww4 = 0.2;
    double ww5 = 0.2;

    DoubleMatrix wR = new Double[,] { {ww1, ww2, ww3, ww4, ww5}};
    DoubleMatrix wc = new Double[,] { { av}, {re}, {itg}, {up}, {cpu} };
    DoubleMatrix wE = new Double[,] { {0.99}, {0.98}, { 0.99 }, {0.99}, {0.97} };
    DoubleMatrix distance = wR * (wE - wc);


    if (distance[0, 0] > 0)
    {
        xxx = 1.0;
        trust = trust + (1 - xxx*trust)/10;
    }
     else
    {
        xxx = Math.Pow(2.0, distance[0,0]);
        trust = trust - (1 - xxx * trust) /10;
    }

    trust = Math.Round(trust, 3);
    dt.Rows.Add(av, re, itg, up, cpu, trust);
```

Figure 16: GlobalTrust Function

The results of each iteration at a periodic interval of time is displayed as output and also appended as a column to the excel worksheet. Output of the code is as follows:



Availability	Reliability	Integrity	Uptime	CPU	Trust
0.99	0.93	0.98	0.99	0.99	0.55
0.99	0.96	0.99	1	0.99	0.505
0.96	0.99	0.97	1	0.99	0.554
0.99	0.99	0.97	1	1	0.509
0.98	1	0.96	1	0.98	0.558
0.99	0.99	0.95	1	0.98	0.602
0.89	0.99	0.96	1	0.95	0.642
0.89	0.94	0.96	1	0.95	0.678
0.87	0.95	0.98	1	0.96	0.71
0.99	0.99	0.98	1	0.99	0.681
0.99	0.99	0.99	1	0.98	0.649
0.99	0.99	0.99	1	0.98	0.614
0.98	0.99	0.99	1	0.97	0.575
0.97	1	0.99	1	0.99	0.532
0.97	1	0.99	1	0.99	0.485

Exported Successfully

Figure 17: Output of Trust Code for CSP

4.5 Comparison with Existing Trust Establishment and Update Technique:

Trust evaluation and update mechanism proposed by [19] was against under the proposed algorithm with 3x SLA parameters i.e. Availability, Reliability and Integrity. The results of which are compared against CSP with 5x SLA parameters proposed in this research i.e. Availability, Reliability, Integrity, Uptime and CPU usage. Graphical comparison is as follows:

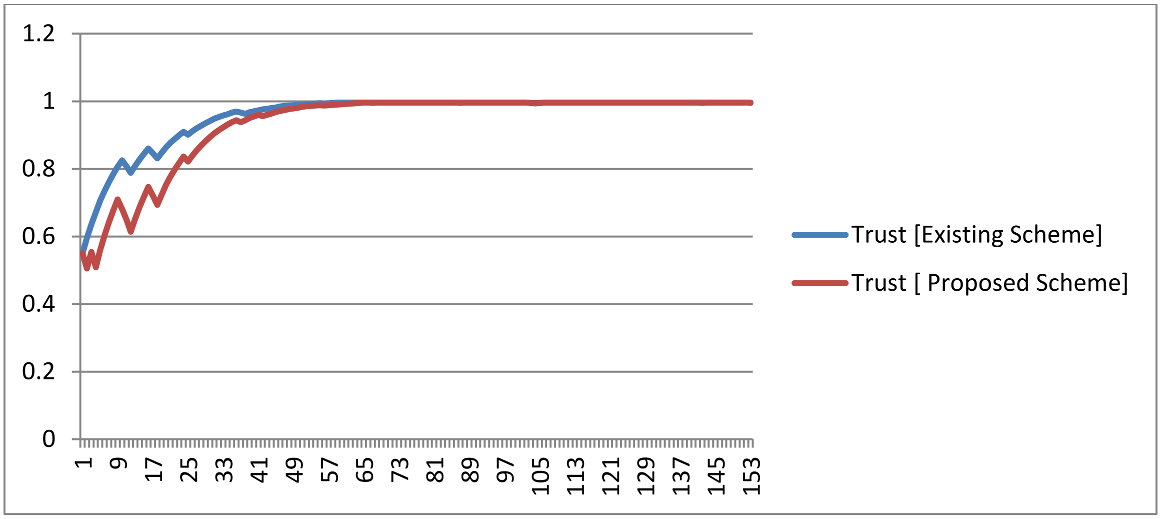


Figure 18: Comparison of Trust Model (Existing vs Proposed)

The graphical results show that the proposed scheme is representing the state of CSP performance in a better way since a larger number of fluctuations and spikes in the value are seen. These spikes reflect the increase/ decrease in value of trust with respect to the increase/ decrease in SLA performance at the CSP end.

4.6 Conclusion

In this chapter, the proposed trust evaluation and update model for federated cloud environment has been discussed and analyzed. The model has been found to be feasible to be implemented practically and gives a more accurate value of trust than the existing scheme.

Chapter 5

CONCLUSION AND FUTURE WORK

This chapter is a conclusion to the presented research work and an identifier to future research directions to be considered. The chapter reviews the presented research work alongwith the prospective points open for consideration for the researchers.

The research has primarily emphasized on update of trust value of a certain CSP participating in a cloud federation in an iterative manner. The trust score is evaluated ans established at each periodic interval and hence the most recent and updated image of a CSP is available to the new CSP who wants to join a federation. The trust value itself is reflected upon by certain QoS parameters defined in the SLA document. Majority of the trust evaluation schemes proposed so far have focused on a maximum of three SLA parameters to establish the trust value. This research has extended the number of SLA parameters of CSP under consideration to five which would give a more appropriated view of cloud performance and includes two more dimensions to analyze the performance of a cloud.

Due to the readily available value of trust, the overhead of re-calculation of trust each time a new CSP wishes to join a federation is eliminated. The model is also unique in the sense that its experimental values have been recorded from a cloud implemented in real-time which is actively providing services to a userbase and comprises of variety of elements i.e. compute, storage, databases, applications and network etc. This gives a better insight than the simulated environment.

The model has proved through its tested value to be an effective approach to establish the trust factor of a CSP with respect to its performance in terms of SLA parameters. The value of trust fluctuates positively as well as negatively as and when the SLA performance increases or decreases respectively.

The research work can be extended in following ways:-

- Increase more SLA parameters to study its effect on trust value. This would give a wider acceptance of the trust established.

- Use trust value to identify malicious and weak cloud service providers. This can remove the troll/ fake entities from cloud environment.
- The proposed model can be enhanced to provide a pricing scheme to the consumers based on the real-time performance of the CSP.

REFERENCES

[1] CloudTweaks, \A history of cloud computing," http://cloudtweaks.com/2011/02/a-history-of-cloud-computing/, 2011, accessed: 2013-01-1.

[2] Personal data in the cloud: A global survey of consumer attitudes. (2017). [eBook] JAPAN: FUJITSU wnloads/SOL/fai/reports/fujitsu_personal-data-in-the-cloud.pdf [Accessed 11 Nov. 2017].

[3] Computer Weekly, A history of cloud computing , "https://www.computerweekly.com/feature/A-history-of-cloud-computing"

[4] NIST Computer Security Resource Center, The NIST Definition of Cloud Computing, "https://csrc.nist.gov/publications/detail/sp/800-145/final"

[5] Apprenda, Cloud Federation, "https://apprenda.com/library/glossary/definition-cloud-federation/"

[6] David Bermbach, Tobias Kurze, Stefan Tai, "Cloud Federation: Effects of Federated Compute Resources on Quality of Service and Cost"

[7] Farzad Sabahi, “Cloud Computing Security Threats and Responses”, 978-1-61284-486-2/11, 2011-IEEE.

[8] D. Chen and H. Zhao, \Data security and privacy protection issues in cloud computing," in *Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on*, vol. 1. IEEE, 2012, pp.647

[9] Priya Govindaraj, N Jaisankar\ A review on various trust models in cloud comptuting, Journal f Engineering Science and Technology Review, 2017

[10] Y. Gil and V. Ratnakar, \Trusting information sources one citizen at a time," in The Semantic WebISWC 2002. Springer, 2002, pp. 162{176.

[11] D. W. Manchala, \E-commerce trust metrics and models," Internet Computing, IEEE, vol. 4, no. 2, pp. 36{44, 2000.

[12] P. Manuel, “A trust model of cloud computing based on Quality of Service”, Annals of Operations Research, Vol.233, Issue.1, pp. 281- 292, 2015

[13] M. Alhamad, T. Dillon, “SLA-Based Trust Model for Cloud Computing”, In the proceedings of the 13th International Conference on Network-Based Information Systems, Japan , pp. 321-324, 2010.

[14] S.Ahmad, B. Ahmad, S. M. Saqib, R. M. Khattak, “Trust model: Cloud’s provider and cloud’s user”, International Journal of Advanced Science and Technology, Vol.44, pp. 69-80, 2012.

[15] L. Zhou and Z. J. Haas, \Securing ad hoc networks," Network, IEEE, vol. 13, no. 6, pp. 24{30, 1999.

[16] K. Aberer and Z. Despotovic, \Managing trust in a peer-2-peer information system," in Proceedings of the tenth international conference on Information and knowledge management. ACM, 2001, pp. 310{317.

[17] A. Tajeddine, A. Kayssi, A. Chehab, and H. Artail, \A comprehensive reputation-based trust model for distributed systems," in Security and Privacy for Emerging Areas in Communication Networks, 2005. Workshop of the 1st International Conference on. IEEE, 2005, pp. 116{125

[18] Zhenhua Tan_, Yicong Niu, Yuan Liu, Guangming Yang ,“A Novel Trust Model Based on SLA and Behavior Evaluation for Clouds” Northeastern University, NEU, Shenyang, P.R. China

[19] Zhen-Hua Tan, Yi-Cong Liu, Nan-Xiang Shi, Xing-Wei, “MCTModel: A Multi-clouds Trust Model Based on SLA in Cloud Computing , Journal of Computers, 2017

[20] Matin Chiregi, Nima Jafari Navimipour , “A new method for trust and reputation evaluation in the cloud environments using the recommendations of opinion leaders' entities and removing the effect of troll entities” Computers in Human Behavior , 2016

[21] Zhenhua Tan_, Yicong Niu, Yuan Liu, Guangming Yang ,“A Novel Trust Model Based on SLA and Behavior Evaluation for Clouds” Northeastern University, NEU, Shenyang, P.R. China

[22] Ayesha Kanwal, Rahat Masood and Muhammad Awais Shibli, “Evaluation and Establishment of Trust in Cloud Federation”, , International Conference on Ubiquitous Information Management and communication (IMCOM, 14’) Columbia January 9-11, 2014

[23] Xiaonian Wua, Runlian Zhanga, Bing Zengb, Shengyuan Zhou , “A trust evaluation model for cloud computing” Information Technology and Quantitative Management (ITQM), 2013

[24] *Mohammed Alhamad* , “SLA-Based Trust Model for Secure Cloud Computing” , 2011

[25] *Eman Aljoumah , Fajer Al-Mousawi , Imtiaz Ahmad , Maha Al-Shammri and Zahraa Al-Jady5*, "SLA in Cloud Computing Architectures: A Comprehensive Study", International Journal of Grid Distribution Computing, 2015

[26] VMware Inc, "vSphere Monitoring and Performance v 6.5", http://www.vmware.com/support/pubs