Trusted Cloud Computing

Submitted By

Vidhika Vasani

16MCEI26

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2018

Trusted Cloud Computing

Major Project

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering(Information & Network Security)

Submitted By

Vidhika Vasani

(16MCEI26)

Guided By

Prof. Vipul Chudasama

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2018

Certificate

This is to certify that the major project entitled **"Trusted Cloud Computing"** submitted by **Vidhika Vasani (Roll No: 16MCEI26)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering of Nirma University, Ahmedabad, is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project part-I and part-II, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof Vipul Chudasama
Guide and Assistant Professor,
CE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Sharda Valevati
Associate Professor and Coordinator,
CE Department
Institute of Technology,
Nirma University, Ahmedabad

Dr. Sanjay Garg
Professor and Head,
CE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Alka Mahajan
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Vidhika Vasani**, Roll. No. **16MCEI26**, give undertaking that the Major Project entitled "**Trusted Cloud Computing**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering(Information & Network Security)** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

_____

Signature of Student

Date: May, 2018

Place: Ahmedabad

Endorsed by

Prof. Vipul Chudasama

(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Prof. Vipul Chudasama**, Assistant Professor, Computer Science Department, Institute of Technology, Nirma University, Ahmedabad for his valuable guidance and continual encouragement throughout this work. The appreciation and continual support he has imparted has been a great motivation to me in reaching a higher goal. His guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

It gives me an immense pleasure to thank **Dr. Sharda Valiveti**, Associate Professor, Computer Science Department, Institute of Technology, Nirma University, Ahmedabad for her valuable guidance and continual encouragement throughout this work.

It gives me an immense pleasure to thank **Dr. Sanjay Garg**, Hon'ble Head of Computer Science and Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his kind support and providing basic infrastructure and healthy research environment.

A special thank you is expressed wholeheartedly to **Dr. Alka Mahajan**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Science & Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Vidhika Vasani

16MCEI26

Abstract

Cloud computing provides services from the available pool of resources. Even with the available condition cloud computing reach peak of success among cloud users. The issue of cloud users face is the barrier of trust between the end-users for using the given services. Conventional security and protection controls keep on being executed on cloud however because of its liquid and dynamic nature, a testable trust estimate of the cloud is required. A crispy input is renewed to the dissimilar members of the related membership functions created on its rate. Since this fact of opinion, the output of a fuzzy logic controller is created on its memberships of the dissimilar membership functions, which can be measured as a array of inputs. This report exhibits an analysis of the present trust administration strategies for cloud operations. In this report proposed a model for Trust administration using Fuzzy Logic, which can help cloud service providers to select trusted resources of datacenter for consumers.

Abbreviations

AL	Authorization Level
ARD	AvailabilityReliabilityData Integrity
AVL	Availability
AWS	Amazon Web Services
CA	Capability
CA	Certification Authority
CP	Certificate Policy
CSP	Cloud Service Provider
CSU	Cloud Service User
DI	Data Integrity
DTCCP	Distributed Trusted Cloud Computing
EL	Entity Level
En	Entropy
He	Hyper entropy
IaaS	Infrastructure as a Service
ID	Identity
IT	Information Technology
IDE	Interface Development Environment
MF	Membership Function
NIST	National Institute of Standards and Technology
OS	Operating System
PaaS	Platform as a Service
PKI	Public Key Infrastructure
QoS	Quality as Service
REL	Reliability
SaaS	Software as a Service
SEP	Stable Election Protocol
SL	Security Level
TaaS	Trust as a Service

TCG	Trusted Computing Group
TCP	Trusted Computing Platform
TCP	Trusted Cloud Computing Platform
TE	Trust Evaluation
TVMM	Trusted Virtual Machine Monitor
VLANs	Virtual Local Area Networks
VM	Virtual Machine

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

Cloud computing is an information technology (IT) model that allows global access to shared pools of configurable system resources and higher-level services that can be quickly provisioned with nominal management effort, often over the Internet. Cloud computing depend on sharing of resources to reach consistency and economies of scale, parallel to a public usefulness. Cloud computing permits creativities to become their applications up and running faster, with less care and upgraded manageability and that it allows IT teams to more quickly alter resources to meet unpredictable and changeable request. Cloud providers naturally use a "pay-as-you-go" model, which can lead to surprising operational costs if managers are not comfortable with cloud-pricing models.

Cloud computing shows the following key features:

- Agility for organizations may be improved, as cloud computing may increase users' flexibility with re-provisioning, adding, or expanding technological infrastructure resources.

- Cost reductions are requested by cloud providers. A public-cloud delivery model converts assets costs (e.g., purchasing servers) to effective expenses. This allegedly lowers barriers to pass, as infrastructure is classically providing by a third party and requirement not be obtained for previous or rare concentrated computing tasks. Estimating on an efficacy computing source is "fine-grained", with usage-based promoting decisions. As well, fewer in-house IT services are essential for execution of developments that usage cloud computing. The e-FISCAL project's state-of-

the-art source covers some training's observing into rate characteristics in more feature, greatest of them final that costs investments depend on the type of events maintained and the category of infrastructure accessible in-house.

- Device and location independence allow operators to access structures using a web browser irrespective of their location or what device they use (e.g., PC, mobile phone). As infrastructure is off-site (typically provided by a third-party) and retrieved by the Internet, operators can attach to it from anyplace.

- Maintenance of cloud computing applications is informal, because they do not requirement to be connected on each operator's computer and can be retrieved from different places (e.g., different work locations, while travelling, etc.).

- Multitenancy allows allocation of resources and costs across a huge pool of operators therefore permitting for:

  - Centralization of infrastructure in locations with minor costs (such as real estate, electricity, etc.)
  - Peak-load volume rises (users requirement not engineer and pay for the resources and gear to chance their maximum possible load-levels)
  - Utilization and effectiveness progress for systems that are repeatedly only 10–20% utilized.

- Performance is observed by IT specialists from the service provider, and dependable and roughly attached architectures are created using web services as the system interface.

- Resource assembling is the provider's computing resources are commingling to serve numerous clients using a multi-tenant model with different physical and resources dynamically allocated and reallocated giving to user request. There is a sense of location individuality in that the customer usually have no switch or information over the location of the provided resource.

- Productivity may be improved when numerous users can effort on the same data concurrently, rather than to come for it to be protected and forwarded. Time may be kept as information does not requirement to be re-entered when fields are

coordinated, nor do users requirement to connect application software promotions to their computer.

- Reliability improves with the use of multiple dismissed locations, which types well-designed cloud computing appropriate for business continuity and tragedy recovery.

- Scalability and elasticity by dynamic ("on-demand") provisioning of resources on a fine-grained, self-service base in near real-time (Note, the VM start-up time differs by VM type, location, OS and cloud providers), without users consuming to engineer for peak loads. This gives the capacity to scale up when the usage requirement rises or down if resources are not existence used.

- Security can progress due to control of data, improved security-focused resources, etc., but fears can continue about loss of switch over certain complex data, and the absence of security for kept kernels. Security is repeatedly as good as or better than other traditional systems, in part because service providers can offer resources to explaining security issues that many clients cannot afford to challenge or which they absence the technical services to address. Though, the difficulty of security is importantly improved when data is spread over a wider area or over a better number of devices, as well as in multi-tenant systems collective by dissimilar users. In addition, user access to security audit logs may be tough or impossible. Private cloud connections are in part inspired by users' want to hold control over the infrastructure and avoid trailing control of information security.

The National Institute of Standards and Technology's description of cloud computing classifies "five important features":

- **On-demand self-service.** A user can individually provide computing abilities, such as network storage and server time, as required repeatedly without needful human interface with each service provider.

- **Broad network access.** Abilities are accessible over the network and opened over ordinary devices that help use by heterogeneous high or profuse customer platforms (e.g., workstations, laptops, tablets, and mobile phones).

- **Resource pooling.** The provider's computing resources are shared to serve numerous clients using a multi-tenant model, with dissimilar physical and virtual

resources dynamically allocated and reallocated giving to customer request.

- **Rapid elasticity.** Abilities can be elastically provisioned and free, in some cases repeatedly, to scale quickly external and inner appropriate with request. To the customer, the abilities obtainable for provisioning frequently appear limitless and can be taken in any amount at any period.

- **Measured service.** Cloud organizations repeatedly switch and improve resource use via leveraging a metering ability at around level of concept suitable to the category of service (e.g., active user accounts, bandwidth, storage and processing). Resource procedure can be supervised, measured, and described, providing transparency for both the customer and provider of the used service.

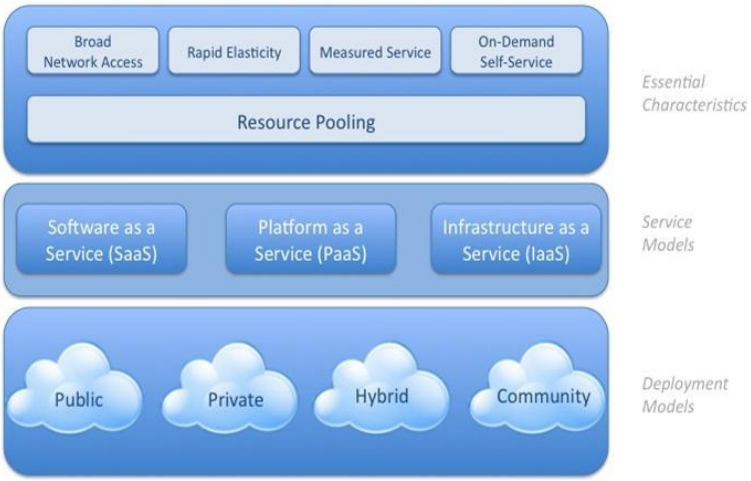


Figure 1.1: NIST Model

However, service-oriented architecture supporters "everything as a service" (with the contractions EaaS or XaaS or simply aas), cloud-computing providers proposal their "services" giving to dissimilar models, of which the three ordinary models per NIST are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). These representations proposal growing concept; they are thus repeatedly represented as layers in a stack: infrastructure-, platform- and software-as-a-service, but these requirements not be associated. For example, one can deliver SaaS applied on physical machines (bare metal), without using primary PaaS or IaaS layers, and equally one can run a program on IaaS and access it straight, without covering it as SaaS.

The NIST's definition of cloud computing describes IaaS as "where the customer is able to install and run random software, which can contain operating systems and

applications. The customer does not achieve or switch the original cloud infrastructure but has switch over operating systems, storage, and installed applications; and maybe incomplete switch of select networking mechanisms (e.g., host firewalls)."

Infrastructure as a service (IaaS) "Infrastructure as a service" (IaaS) denotes to online services that deliver high-level APIs used to deference numerous low-level facts of original network infrastructure similar physical computing resources, location, data subdividing, ascending, security, backup etc. A hypervisor, such as Oracle VM, Oracle VirtualBox, Xen, KVM, Hyper-V, or VMware ESX/ESXi, LXD, runs the virtual machines as visitors. Pools of hypervisors inside the cloud effective system can provision huge numbers of virtual machines and the capability to measure services up and down giving to clients' changing supplies. Linux containers run in remote barriers of a single Linux kernel consecutively straight on the physical hardware. Linux control groups and namespaces are the original Linux kernel technologies used to separate, safe and achieve the containers. Containerization proposals higher performance than virtualization, because there is no hypervisor above. Also, container volume auto-scales dynamically with computing load, which removes the problem of over-provisioning and allows usage-based promoting. IaaS clouds repeatedly proposal extra resources such as a raw block storage, file or object storage, virtual-machine disk-image library, IP addresses, firewalls, load balancers, software bundles, and virtual local area networks (VLANs).

IaaS-cloud providers supply these resources on-demand from their large pools of equipment installed in data centers. For wide-area connectivity, clients can use either the Internet or carrier clouds (dedicated virtual private networks). To deploy their applications, cloud users install operating-system images and their application software on the cloud infrastructure. In this model, the cloud user covers and keeps the operating systems and the application software. Cloud providers classically bill IaaS services on an efficacy computing basis: cost reproduces the amount of resources spent and assigned.

Platform as a service (PaaS) The NIST's definition of cloud computing defines Platform as a Service as:

The ability providing to the customer is to organize onto the cloud infrastructure consumer-created or acquired applications created using libraries, tools, services, and programming languages supported by the provider. The customer does not achieve or switch the original cloud infrastructure together with network, storage, operating systems,

or servers, but has control over the installed applications and maybe configuration settings for the application-hosting environment.

PaaS retailers proposal a growth environment to application developers. The provider classically grows toolkit and ethics for development and channels for delivery and expense. In the PaaS models, cloud providers send a computing platform, classically with operating system, programming-language execution environment, web server, and database. Application developers can grow and run their software results on a cloud platform without the cost and difficulty of ordering and handling the original software and hardware layers. With some PaaS proposals like Microsoft Azure, Oracle Cloud Platform and Google App Engine, the original computer and storage resources scale repeatedly to match application request so that the cloud operator does not have to assign resources physically. The latter has also been planned by an architecture pointing to simplify real-time in cloud environments. Even more specific application types can be provided by PaaS, such as media programming as provided by services like bitcodin.com. Some integration and data management providers have also contained specialized applications of PaaS as delivery models for data solutions.

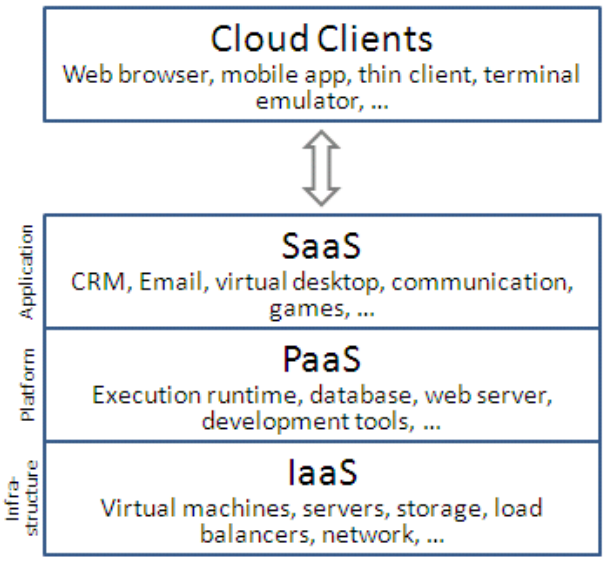


Figure 1.2: Cloud Computing Layers

Software as a service (SaaS) The NIST's definition of cloud computing defines Software as a Service as:

The ability provided to the client is to use the provider's applications running on a cloud infrastructure. The applications are available from many consumer devices finished

either a thin client interface, such as a program interface or a web browser (e.g., web-based email). The client does not achieve or switch the original cloud infrastructure with network, individual application abilities, operating systems, servers, or even storage, apart from imperfect user-specific application configuration settings.

In the software as a service (SaaS) model, users advantage access to application software and databases. Cloud providers achieve the infrastructure and platforms that run the applications. SaaS is sometimes mentioned to as "on-demand software" and is typically estimated on a pay-per-use base or using a payment fee. In the SaaS model, cloud providers deploy and operate application software in the cloud and cloud users access the software from cloud consumers. Cloud users do not achieve the cloud infrastructure and platform where the application runs. This removes the requirement to deploy and run the application on the cloud user's own computers, which make simpler keep and provision. Cloud applications change from other applications in their scalability—which can be attained by replicating tasks onto multiple virtual machines at run-time to happen altering work request. Load balancers allocate the effort over the set of virtual machines. This development is clear to the cloud operator, who realizes only a single access-point. To accommodate many cloud operators, cloud applications can be multitenant, meaning that any machine might help more than one cloud-user organization.

The estimating model for SaaS applications is classically a monthly or yearly level payment per user, so amounts develop accessible and adaptable if users are added or removed at any point. Supporters privilege that SaaS gives a business the possible to decrease IT operational charges by outsourcing hardware and software keep and provision to the cloud provider. This allows the business to move IT operations costs gone from hardware/software costs and from personnel costs, towards assembly other goals. In addition, with applications presented centrally, updates can be unrestricted without the essential for users to fix new software. One disadvantage of SaaS derives with storage the users' data on the cloud provider's server. As a result, there could be illegal access to the data.

There are three deployment clouds in cloud computing which is describe in detail.

1. **Private cloud**

   Private cloud is cloud infrastructure worked uniquely for a single organization, whether succeeded within or via a third-party, and presented either inside or out-

side. Responsibility a private cloud development needs important engagement to virtualize the business environment and needs the organization to re-evaluate results about current resources. It can recover business, but each step in the development increases safety problems that must be talked to avoid thoughtful vulnerabilities. Self-run data centers are usually capital exhaustive. They have an important physical footprint, demanding distributions of space, hardware, and environmental controls. These properties must be restored occasionally, subsequent in supplementary capital costs. They have involved disapproval because operators "still have to buy, build, and manage them" and therefore do not advantage from less hands-on management, basically "the economic model that makes cloud computing such an interesting concept".

2. **Public cloud**

   A cloud is named a "public cloud" when the services are reduced over a network that is exposed for public use. Public cloud services might be permitted. Theoretically there may be slight or no modification among public and private cloud architecture, though, security thought may be significantly dissimilar for services (applications, storage, and other resources) that are completed accessible by a service provider for a public audience and when statement is realized over a non-trusted network. Usually, public cloud service providers like Amazon Web Services (AWS), Oracle, Microsoft and Google own and activate the infrastructure at their data center and access is usually by the Internet. AWS, Oracle, Microsoft, and Google also proposal through connect services called "AWS Direct Connect", "Oracle Fast Connect", "Azure Express Route", and "Cloud Interconnect" correspondingly, such connections essential clients to purchasing or agreement a private connection to a peering point accessible via the cloud provider.

3. **Hybrid cloud**

   Hybrid cloud is a structure of two or more clouds (private, community or public) that continue dissimilar objects but are certain organized, contribution the benefits of numerous arrangement models. Hybrid cloud can also mean the capability to connect association, succeeded and/or dedicated services with cloud resources. Gartner

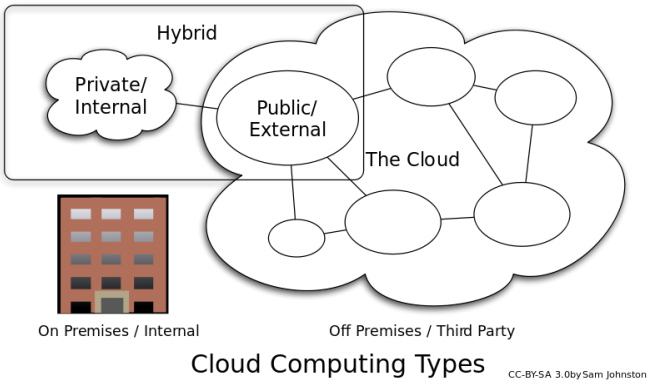


Figure 1.3: Deployment Clouds

describes a hybrid cloud service as a cloud computing service that is collected of some mixture of private, public and community cloud services, from dissimilar service providers. A hybrid cloud service marks separation and provider limits so that it can't be purely put in one group of private, public, or community cloud service. It permits one to spread either the volume or the ability of a cloud service, by collection, combination or customization with another cloud service.
Wide-ranging use cases for hybrid cloud configuration be. For example, an organization might supply complex consumer data in house on a private cloud request but intersect that application to a business intelligence application provided on a public cloud as a software service. This example of hybrid cloud spreads the abilities of the originality to distribute an exact business service over the calculation of externally accessible public cloud services. Hybrid cloud implementation depends on a few features such as data security and compliance requirements, level of control required over data, and the applications an organization uses.

Additional example of hybrid cloud is one wherever IT organizations use public cloud computing resources to happen brief volume desires that cannot be happened by the private cloud. This ability allows hybrid clouds to service cloud overflowing for mounting across clouds. Cloud overflowing is an application deployment model in which an application innings in a private cloud or data center and "bursts" to a public cloud when the demand for computing volume rises. A main benefit of cloud overflowing and a hybrid cloud model is that an organization earnings for additional compute resources only when they are required. Cloud overflowing al-

lows data centers to generate an in-house IT infrastructure that provisions normal workloads, and use cloud resources from public or private clouds, through points in processing difficulties. The model of hybrid cloud, which is constructed on heterogeneous hardware, is called "Cross-platform Hybrid Cloud". A cross-platform hybrid cloud is generally powered by dissimilar CPU architectures, for example, x86-64 and ARM, underneath. Users can clearly install and measure applications without data of the cloud's hardware variety. This kind of cloud occurs from the increase of ARM-based system-on-chip for server-class computing.

1.2 Introduction of trusted cloud computing

Cloud computing infrastructures allow companies to cut costs by outsourcing computations on-demand. Though, consumers of cloud computing services presently have no resources of proving the privacy and honesty of their data and computation. To address this problem the design of a trusted cloud computing platform (TCCP) was presented. TCCP allows Infrastructure as a Service (IaaS) providers such as Amazon EC2 to proposal a closed box performance environment that guarantees private performance of visitor virtual machines. Likewise, it permits users to show to the IaaS provider and control whether the service is secure before they promotion their virtual machines. The goal of trusted cloud computing is to produce the addition of virtual machines private which is installed by the service provider.

Clients can confirm that the addition is private and avoid review of addition state at the service provider site. It permits clients to confirm that addition is protected and installed with support of the cloud provider. Two components: A trusted co-ordinator and A trusted virtual machine monitor (TVMM). It services to control whether the service is protected before they promotion their VM. Hence, TCCP offers a closed box implementation environment by covering the concept of trusted platform to a whole IaaS back-end.

Cloud computing offers public a technique to share large quantity of spread resources belonging to dissimilar organizations. That is respectable way to share many types of spread resources, but it also makes safety complications more confuse and more significant for operators than earlier. We analyze some security requirements in cloud computing environment. The security difficulties both in hardware and software, we provided a

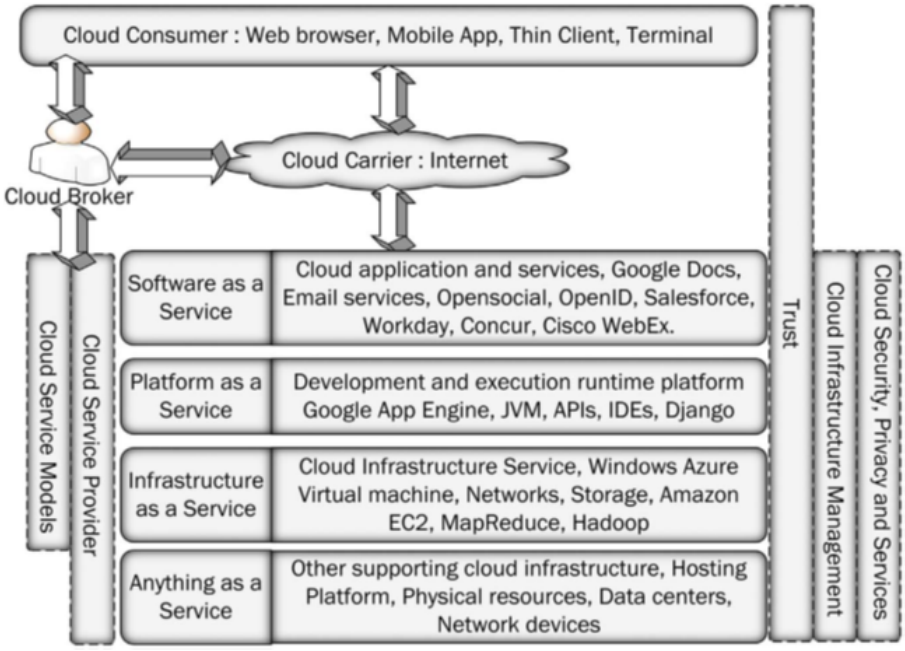


Figure 1.4: Trust in cloud
[1]

method to build a trusted computing environment for cloud computing by integrating the trusted computing platform(TCP) into cloud computing system.

Trusted computing is a wide term that denotes to technologies and offers for deciding computer security difficulties over hardware improvements and related software changes. Several major hardware productions and software sellers, together known as the Trusted Computing Group(TCG). The TCG progresses and helps condition for the safety of computer resources from pressures modelled by malicious objects without infringing on the privileges of end users.

Into figure 1.4 shown trust in cloud. To understand what is trust and security issues in cloud first we known about the NIST framework which is described in detail earlier. So into the figure we can shown that trust is given connection between cloud infrastructure management and cloud services model. Cloud service provider, cloud consumer, cloud carrier and cloud broker are connected to each other.

Chapter 2

Literature Survey

In modern area of cloud computing technologies,Trust is used in vast area of applications. The important barrier to general approval of cloud computing is the absence of trust in clouds by possible clients. In present scenario trust depends mainly on awareness of reputation, and self-assessment via providers of cloud services.

2.1 Different techniques for trusted cloud computing

Here, a list of literature survey papers which have worked on the different techniques of Trusted cloud computing are described in table:

Paper	Synopsis	Conclusion
TrustCloud: A Framework for Accountability and Trust in Cloud Computing[2]	This paper integrates main issues and challenges in realizing a trusted cloud over the use of investigator controls, and offerings the TrustCloud framework, which talks accountability in cloud computing through methodological and policy-based approaches.	The result grows a system created on the TrustCloud framework that build cloud users a single point of view for accountability of the CSP.

Paper	Synopsis	Conclusion
Trust Management in Cloud Computing: A Survey[3]	Study about the existing trust management methods about to the performance of cloud service providers taking into thought other characteristics like privacy, credibility, security, user feedback, etc.	Recommendation created trust and reputation created trust are benefited in expecting to the performance of a cloud service provider.
Trusted Cloud Computing with Secure Resources and Data Coloring[4]	Exploring the techniques which are given protection multi-way verification, allow single sign-on in the cloud, and tighten access control for complex data in both public and private clouds.	The proposed reputation system and data-coloring mechanism to defend data-center access at a coarse-grained level and protected data access at a fine-grained file level.
Trust mechanisms for cloud computing[5]	Describe limits of creating trust by offering more difficult mechanisms based on characteristic certification, evidence, and authentication.	Develop trust mechanisms for cloud computing in five groups– SLA verification based, reputation-based, trust as a service, transparency mechanisms, and audit, formal accreditation, and standards.
Credibility-Based Trust Management for Services in Cloud Environments[6]	Detect the malicious trust feedback from attackers with credibility model and replication determination model which are dynamically decides among credible trust responses and best replica amount of the trust management service.	Develop trust management service which can be continuously kept at a desired availability level and the methods have been authenticated by the prototype system and experimental results.

Paper	Synopsis	Conclusion
CloudArmor: Supporting Reputation-Based Trust Management for Cloud Services[7]	CloudArmor framework which is a reputation-based trust management technique. Which offers a set of functionalists to transport trust as a service (TaaS).	This method given the highly active, spread, and non-transparent nature of cloud services, management and creating trust between cloud service users and cloud services remains a important challenge.
Trust Modeling in Cloud Computing[8]	This paper use some security mechanisms that allow cloud service end users to estimate the trust level of numerous cloud services and resources based on fuzzy theory on a Eucalyptus cloud platform.	Estimate cloud trustworthiness using different methods.
A Distributed Approach towards Trusted Cloud Computing Platform[9]	Discuss about dissimilar infrastructure level attacks and offer a distributed result to implement it with the benefit of trusted cloud computing platform.	Distributed Trusted Cloud Computing Platform (DTCCP) which performances as an oracle platform to update the cloud user that the platform on which they demand to run their computations is truly trusted or not.
A Trust Computing Mechanism For Cloud Computing[10]	Propose a trust design and development mechanism which is used to measure the performance of cloud systems and expresses trust scores for different service level supplies.	The planned mechanism has been verified under a virtual environment and the results have been obtainable.

Paper	Synopsis	Conclusion
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[11]	Recommend a technique for opinion leaders and troll entity identification via three topological metrics, with input-degree, output-degree and reputation measures also calculated trust value with the help of five characteristics of resource.	This paper assesses the trust by since the effect of opinion leaders on other entities and eliminating the troll entities' effect in the cloud environment.
Trust Management Tools for Internet Applications[12]	Trust management is the action of gathering, organizing, analyzing and offering suggestion relating to capability, trustworthiness, security or reliability with the resolution of assembly calculations and results about trust associations for Internet applications.	For trust provisions using SULTAN trust management toolkit which is monitoring trust in cloud.

Table 2.1: A list of survey papers on different techniques for trusted cloud computing

2.2 Fuzzy logic using in trusted cloud computing

Fuzzy logic is a method of many-valued logic in which the truth values of variables can be any real number among 0 and 1. It is working to handle the idea of limited truth, where the truth value can array among entirely true and entirely false.

Here, a list of literature survey papers which have worked on trusted cloud computing using fuzzy logic are described in table:

Paper	Synopsis	Conclusion
Estimating Trust Value for Cloud Service Providers using Fuzzy Logic[13]	In this paper a model for Trust Management based on Fuzzy Logic has been established, which can benefit clients make a knowledgeable choice to choosing the suitable CSP as each their condition.	Provide a diversity of strategies to dissimilar types of clients and rating dissimilar CSPs.
SEPFL routing protocol based on fuzzy logic control to extend the lifetime and throughput of the wireless sensor network[14]	Fuzzy logic control created on three variables, bulk of nodes, distance of nodes forms improper station, and the battery level of nodes laterally with the traditional edge values used in Stable Election Protocol (SEP) are used to improve the process of cluster head selection in the present SEP protocol and expand the time and amount of the Wireless Sensor Network.	Nodes with high outstanding energy, high possibility, and demanding less energy for interactive with other nodes as well as the base station are chosen as the cluster heads.
A Trust Model in Cloud Computing Based on Fuzzy Logic[15]	Use fuzzy logic estimate trust value of a provider in cloud setting, therefore growing the efficiency of the system.	The model gives the operators an impression of trustworthiness of a cloud resource provider.
VM consolidation approach based on heuristics, fuzzy logic, and migration control[16]	To complete energy-QoS balance in cloud using fuzzy logic and heuristic based virtual machine association method.	The paper offers energy efficient algorithms that can minimize energy consumption while care the quality of service (QoS) at a satisfactory level.

Paper	Synopsis	Conclusion
Fuzzy logic based job scheduling algorithm in cloud environment[17]	The paper method simplifies the difficulty of the algorithm and decreases the overhead related with choosing suitable and defensible virtual machine for a certain task.	It uses fuzzy logic to regulate the general prospect vector of the task based on the equality of the provision of resource.
A Fuzzy Logic-based Controller for Cost and Energy Efficient Load Balancing in Geo-Distributed Data Centers[18]	The paper works on a fuzzy logic-based algorithm for rate and energy effective load balancing between numerous data centers of a cloud service provider.	Goal of this paper is using recreation of a case study calculated giving to renewable energy sources, real-world traces of workload, and electricity market prices presented that proposed method is bright to meaningfully decrease the cost of the cloud provider.
Modeling Fuzzy Scheduling in Infrastructure as a Service Cloud[19]	This work shows the outcomes of lease scheduling in the Infrastructure as a Service (IaaS) Cloud in the undefined conditions, and usage of Fuzzy Logic for the similar.	The goal is the scheduling rules defined by the IEC standard is changed and final defuzzifed values have been used for lease scheduling in IaaS Cloud.

Table 2.2: A list of survey papers on fuzzy logic using in trusted cloud computing

Chapter 3

Problem Statement

The study interested in trust in cloud computing has exposed some motivating drifts.

- Operators do not trust existing systems
- The government cannot solution trust problems
- Education is key to trust
- Trust cannot be acquired

Defendants obviously do not trust existing systems in place but resoundingly use them anyhow. Education of the operator is very imperative to creating trust. In simple terms, how the CSP is successful to safe their data. It is up to the CSP to be the educator to receive that trust. Presently, of the top CSPs, it is tough to find confidentiality policies and direct through them. Trust cannot be accepted. It is the invention of a relationship over time.

3.1 Existing System

Here discuss about existing systems in detail. There are mainly discussion about Data coloring and Software watermarking system, Public key infrastructure system and Profile identity management system.

1. **Data coloring and Software watermarking:** The architecture usages data coloring at the data object level or software file. These agreements separate operator access and protect complex information from provider access, as Figure 3.1 shows.[4]

Data Coloring and Software Watermarking certain cloud computing's use of common files and datasets, a challenger could cooperation security, copyright and privacy in a cloud computing environment. In cloud need to effort in a trusted software environment that offers beneficial tools for structure cloud applications over threatened datasets. Noted this model to increase exclusive data colors to defend huge datasets in the cloud.

In this method study cloud security, a public property. To guard it, association the benefits of software watermarking and protected cloud storage over trust cooperation and data coloring. Figure 3.1 shows the datacoloring idea. The woman's image is the data entity presence secured. Figure 3.1a illustrations the forward and backward color group developments. Then improve the cloud drops (data colors) into the contribution photo (left) and eliminate color to re-establish the unique photo (right). The coloring method usages three data features to produce the color: the predictable value (Ex) be contingent on the data content, while hyper entropy (He) and entropy (En) enhance unpredictability or indecision, which are self-determining of the data satisfied and identified only to the data owner. Communally, these three purposes produce a group of cloud drops to form a single "color" that other cloud users or providers can't notice. In this model also, usage data coloring at changing security levels built on the variable charge function practical.

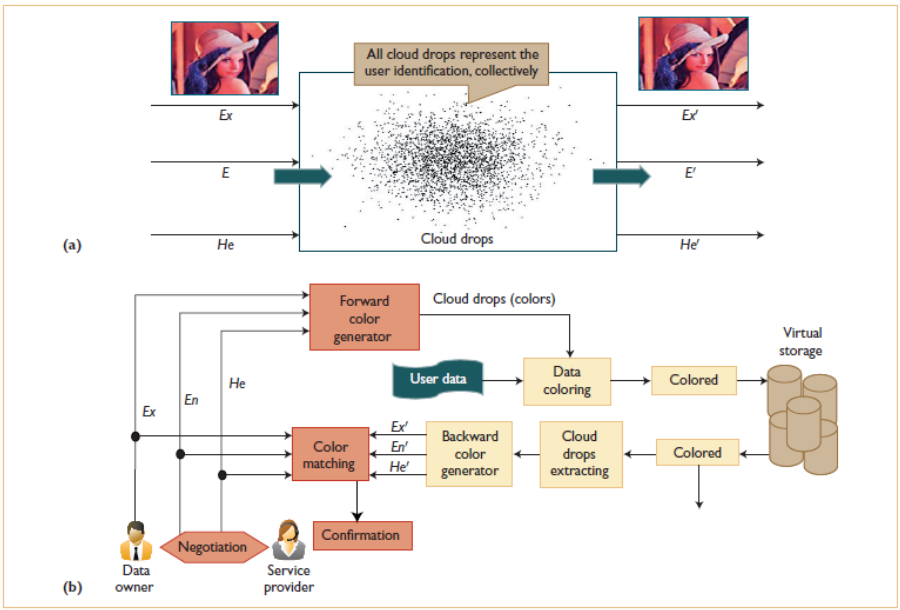


Figure 3.1: Data coloring and Software watermarking using type-2 fuzzy logic [4]

The method can put on to defend relational databases, images, software, video, and documents. Figure 3.1b illustrations the complex in the color-matching procedure, which goals to secondary a colored data object with its holder, whose user documentation is also colored with the equal Ex, He and En identification features. The color-matching procedure guarantees that colors practical to user credentials match the data colors. This can recruit numerous trust-management procedures, including verification and approval. Virtual storage provisions inserting, color generation, and removal. Merging protected data coloring and data storage, it can avoid data objects from existence changed, stolen, or damaged, removed. Therefore, genuine users have only access to their wanted data objects. The computational difficulty of the three data features is much minor than that achieved in predictable encryption and decryption designs in PKI services. The watermark-based system therefore invites a very little above in the coloring and decoloring procedures. The He and En meanings' chance assurances data owner confidentiality. These features can exclusively decide dissimilar data objects. Providers can implement this planned standing system and data-coloring mechanism to defend data-center access at a coarse-grained level and protected data access at a fine-grained file level.

2. **Public Key Infrastructure:** The "formal" trust mechanisms are need in cloud computing. In a correlated domain, PKI is a broadly used established technology that services "formal" trust mechanisms to provision validation, digital signature, and key certification as well as characteristic certification and authentication.

   To shorten the discussion, study the example shown in Figure 3.2 Alice takes a digital file allegedly signed via Bob with his private key K'b. To authenticate, she wants Bob's public key Kb. Accept that Alice trusts individual her trust presenter certification expert CA1, and she identifies only K1, her trust presenter's public key. For her to confirm the signature on the file as being Bob's, she wants to determine a certification path from CA1 to CA3 who has delivered Bob's public key certificate. As exposed in the figure, Alice usages CA1's public key K1 to authenticate CA2's public key K2; since Alice trusts CA1 on public key certification, and CA2's public key is certified by CA1, Alice can trust that CA2's public key is K2; then Alice uses K2 to authenticate CA3' public key K3; and lastly uses K3 to authenticate Bob's public key Kb. The key problem is why Alice should trust K3

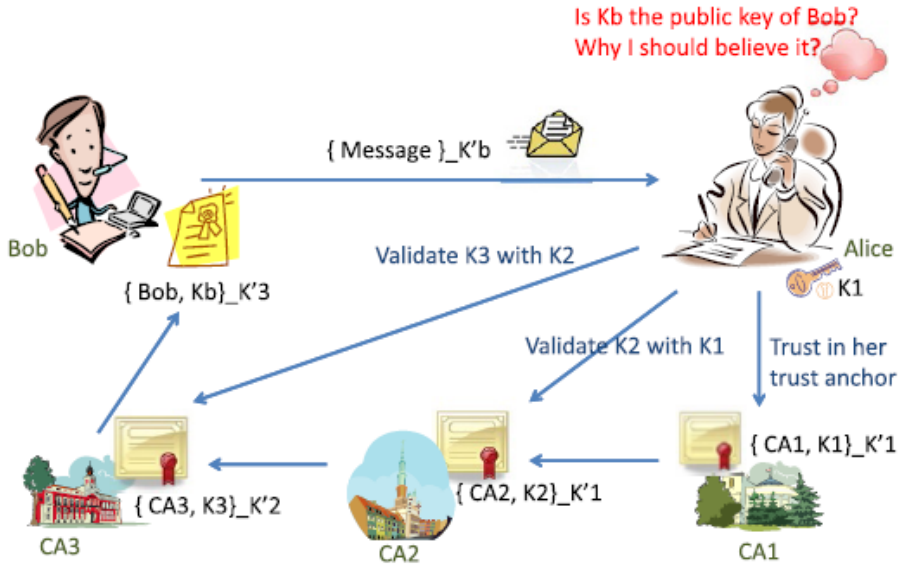


Figure 3.2: Public Key Infrastructure
[5]

is CA3's public key and Kb is Bob's public key? Basically, to conclude confidence in a declaration "Bob's key is Kb", Alice wants to trust CA3, the inventor of that declaration, with respect to the truth of the declaration; though, this increases questions that request about the basis of that trust, and how the trust is indirect or considered. Some research proposes that the trust derives from recommendations laterally the chain of certificates via those certificate issuers; but the preparation of digital certification and authentication in real PKI systems proposes that the trust derives from submission with sure certificate policies. As quantified in IETF RFC 5280, in accumulation to the basic declaration that binds a public key with a focus, a public key certificate also covers a certificate policy (CP) allowance. For a public key certificate allotted to a CA, the certificate means that the distributing CA who follows to the itemized CP declares that the issue CA has the certified public key, and the issue CA also observes to the itemized CP. As a outcome, to conclude Alice's trust in CA3's key and Bob's key, she requirement trust that CP in the intelligence that any CA compatible to that CP will create legal public key certificates. There are additional composite and motivating issues in PKI trust. In rapid, as PKI is presently expert, trust in a certification authority (CA) with respect to distributing and keeping legal public key certificates is created on the CA's conformance with positive certificate rules. Certificate rules show a essential

part in PKI trust. This trust mechanism calls as policy-based trust.[5]

3. **Profile identity management system:**Profile or operator personalizing is not an innovative concept in computing world. Currently personalizing has broadly been used to initial content and facilities giving to the user's benefits and preferences. The personalizing characteristics are used to current users with satisfied or facilities that tie their profiles. In the cloud computing environment, it is extremely needed to have nearly generous of involuntary mechanism in place that can confirm the access and security rules of resources and services for the cloud users. In this explanation, researcher used a profile identity management system that can be used to describe access and security rules for the resources and services accessible on the cloud. The profile in this case is an object that has some prescribed rights to access cloud services.

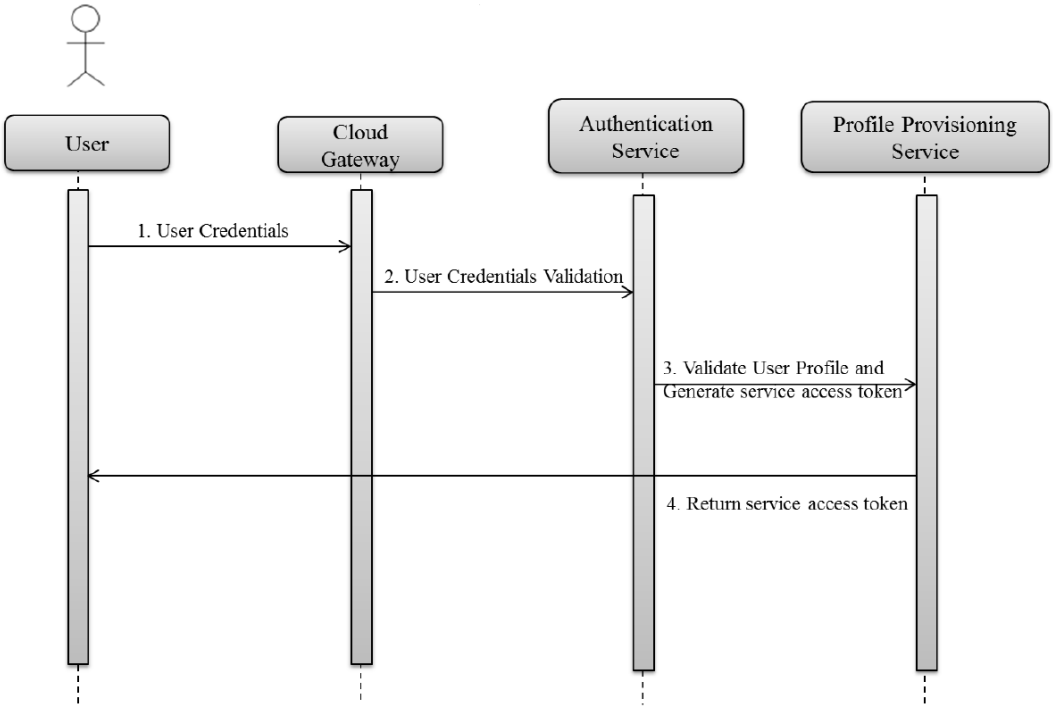


Figure 3.3: Profile Identity Management
[20]

Once a user accesses the cloud and delivers the identifications, the certification system in place legalizes the user identifications and authenticates the user profile. When the user profile is authenticated the access, token is approved to the user for all the services that are accessible to the user of that specific profile. This avoids user to perform repetitive permissions for each service. Similarly, it expands the

general security of the cloud as only services which are accessible for that specific profile are visible to the user. The high-level process diagram of this system is existing in Figure 3.3 As a primary phase in the process, the user refers his/her identifications to the cloud gateway, wherever the verification service will legalize the user. When the user is legalized, the profile provisioning service legalizes the profile, and creates a user's service access token. Later, this token is collective with the resource provisioning service and is also approved to the user in a verification access note. The advantage of distribution the access token is that operators accordingly remove the procedure of user verification for each service they need to access. The access token deceases when the user cyphers out from the cloud or once the access assembly deceases.

3.2 Drawback-Gap

The trust relation is designed by the trustee that is the trusting party and the trustee that is the party which is to remain trusted. The building relation for trust is based on the trustworthiness of the trustee to performance in the greatest attention of the trustee and grade of trust that the trustee spaces on the trustee. Trust is the most multipart relationship between objects because it is individual and tough to be estimated. Trust models are measured as an organization that benefits to estimation trust on the CSP's or the third-party providers that are providing the cloud services. The cloud service users (CSU) always must keep trust on the cloud service providers (CSP) and the CSPs must keep trust on the CSUs for a strong creation of cloud services. In the cloud computing scenarios, the CSUs give all their digital resources in the hands of the CSPs and the CSPs hold direct control over nearly all the security factors. This is the reason why a CSP must consult a proper equal of trust to the CSUs. The component of trust is an important element in the wide use and operation of cloud services. The relationship of trust is established between the two parties which are stated with both party. An entity should provide belief confidence, ability and integrity to another entity to build trust. There are some issues of trust between CSP and Datacenter which are categorized by considering following factors.

1. Is it possible for CSP to evaluate trusted resources of datacenter?

2. Does CSP provide trust interaction which is useful for cloud services in datacenter?

3. Due to uncertainty in trust various administration operation how CSP will provide trust rating for datacenter?

To address the above question model of trust with help CSP to give relation between trust interaction and trust management service. Following figure shown services layer and service request layer. In service layer cloud services and Trust management services are available which are connected through trust interaction. Here shown model of the Trust administration framework into the Figure 3.4.

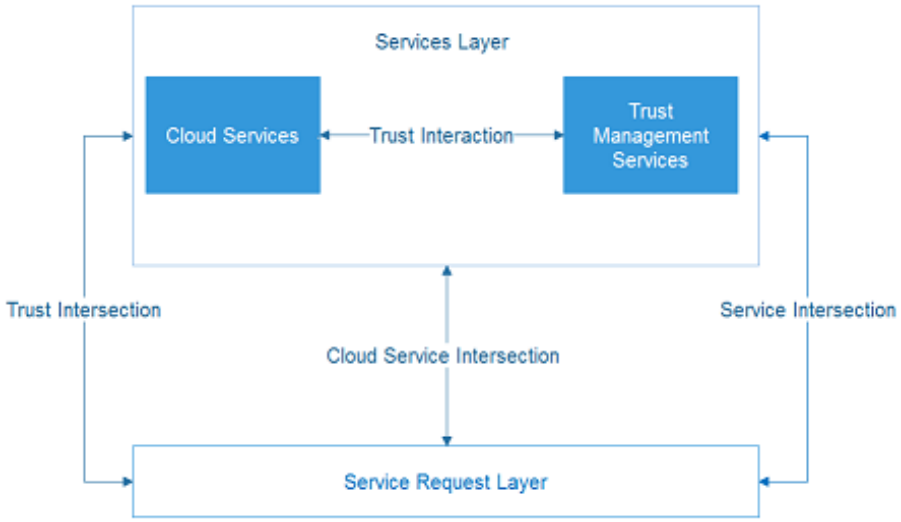


Figure 3.4: Trust administration framework

The resource with pay peruse should be highly accessible with trust relation. These novel structures offer the resources to control huge substructures corresponding datacentres over virtualization otherwise occupation supervision then resource supervision.Data creation remains a feature of cloud then this occurs trendy a method that might contain several gatherings then remains not measured through the data holders. CSPs confirm accessibility through repeating data in various datacenters. It remains tough toward assurance that a duplicate of the data or its backups remain not kept otherwise managed trendy a firm control, or that entirely these duplicates of data remain removed uncertainty such a demand remains finished.

A key modification is that through cloud computing it can be tough, or even terrible, to classify faithfully where the organization's data essentially is. This remains comparatively because CSPs could take server fields in some nations, then it might not remain conceivable used for the CSP to assurance to a client that data resolve remains managed

in a server field, or even nation. Amazon Web Services (AWS) then Google take several data centres global, details of the places of which are repeatedly private. The trust is recognized in a centralized storage area. In many methods, it desires to remain trusted the third party to effort on the data. Since the operator can run the data separately from the ratings they provide. Meanwhile the data remains spread between the objects, it stands actual tough to reserve the confidentiality in the distributed model. The global Reputation-based trust rating remains considered through a usual of several connections.

Chapter 4

Proposed Solution

In this section we proposed a method to calculate trust in cloud. In cloud serve trust as a rate to the resources of datacenter. Using resources, we have values of CPU utilization, user identification and job status. Now with these values we can find equation for characteristics of resources. Characteristics of resources are Availability, Reliability, Data Integrity, Identity and Capability. In cloud computing datacenter cover at an IaaS level. It is a huge field and it is depending on host, host is depending on VMs. If we provide Trust as a Service(TaaS) to the clients from the datacenter firstly CSP rated to the resources of datacenter. For giving a rate to the huge resources here used fuzzy logic technique. Which is implement in MATLAB tool and generate FIS file. Next step is implement FIS file into opensource simulator which is shown the rated resources.

In figure 4.1 shown architecture of datacenter which shown connection between resources and datacenter. In this figure shown hypervisor and VMs which are bellowing to the datacenter.

4.1 Proposed Model

In proposed model using Mamdani fuzzy interface system for give rate to the resources of datacenter. Using characteristics of resources and dataset of 2000 resources. Further implement in MATLAB tool. In figure 4.2 we can show the proposed model.

4.2 Characteristics of resources

1. **Availability (AVL):** The data storage and provision resources are defined by AVL and it remains made of serviceable and reachable request by approved object. It

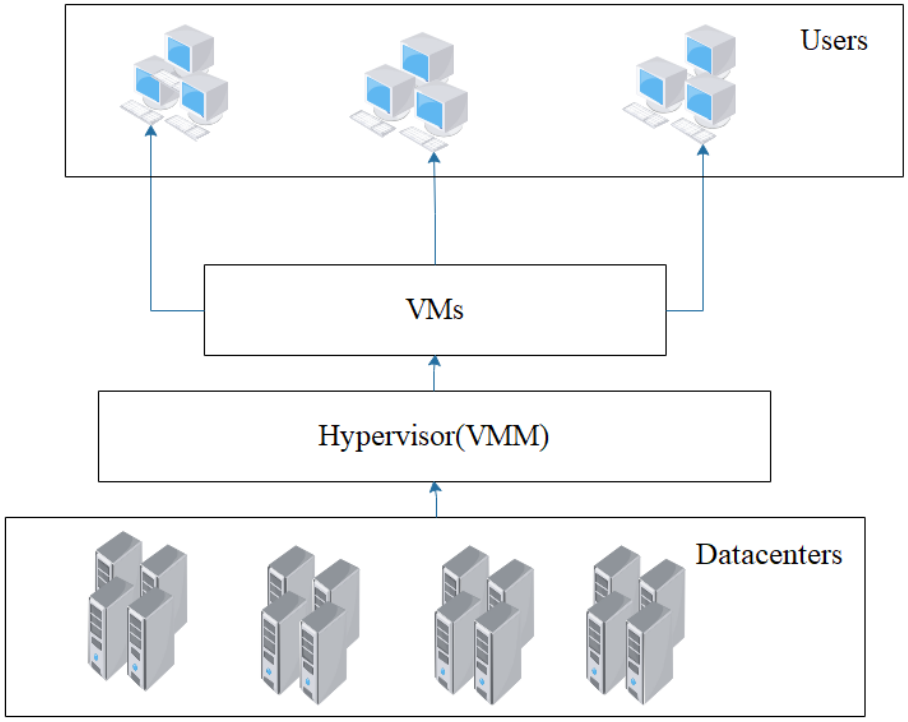


Figure 4.1: Architecture of datacenter

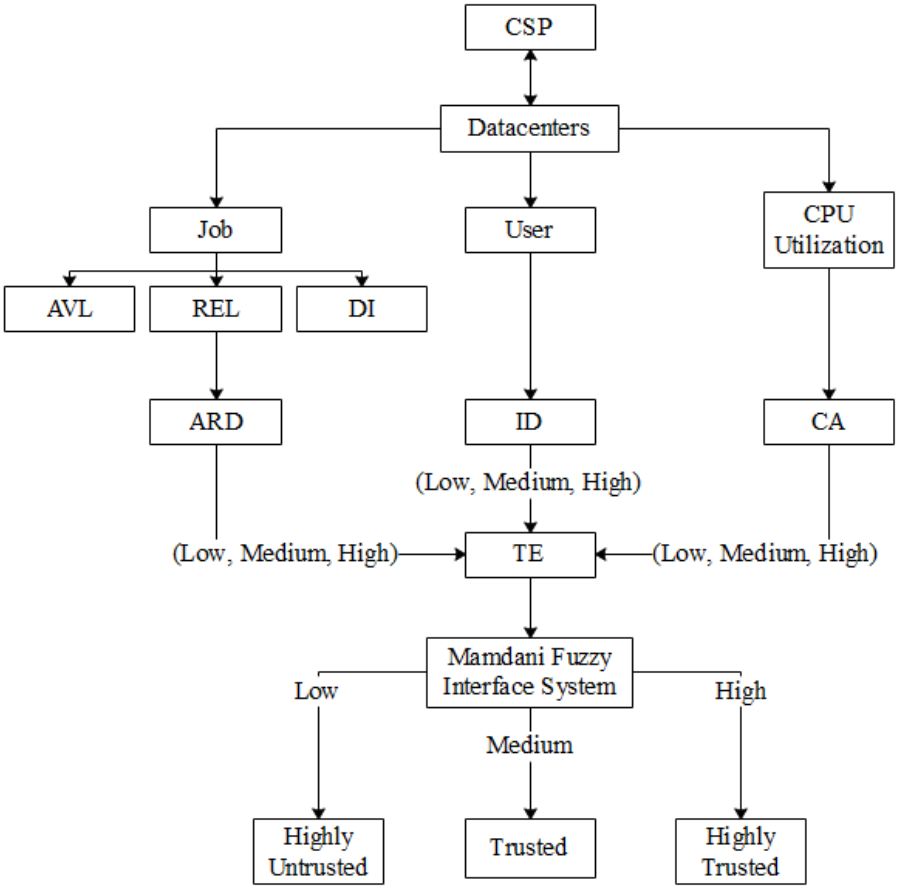


Figure 4.2: Proposed Model

means that the services are presented even many nodes undergoes failure. It is the related to time of a system or component for the usefulness of total time which is mainly essential to purpose. The AVL of resources r is estimated via eq(4.1).

$$AVLr = Ar/Nr \tag{4.1}$$

For Nr means the total no. of jobs given to the resource r, Ar means the total no. of jobs taken by the the resource r.

2. **Reliability (REL):** For trust REL is an important factor. It denotes to the capacity of a software element to dependably make affording to its provisions. The REL of resources r is estimated via eq(4.2).

$$RELr = Cr/Ar \tag{4.2}$$

For Cr means the total no. of jobs completed by the resource r, Ar means the total no. of jobs taken by the resource r.

3. **Data Integrity (DI):** Security is a main problem that desires different consideration in the clouds. The term Data Integrity stands for confidentiality and accuracy of the information. The trust or assurance about executor's behaviour is given by the honesty of the executor. Data protection and data precision are included by security. Information cost capacity occur due to poor network inactivity. Accuracy damage capacity occur due to superseded calculating infrastructure. The DI of resources r is estimated via eq(4.3).

$$DIr = Dr/Cr \tag{4.3}$$

For Cr means the toal no. of jobs completed by the resource r, Dr means the total no. of jobs data integrity preserved by the resource r.

4. **Identity (ID):** Identity of users checking through different levels of security. The level of the cloud service user has been confidential into the subsequent levels: Entity

Protection level, Authorization level and Security level. This level differs since 0 to 1. The ID of resources r is estimated via eq(4.4).

$$IDr = ALr + ELr + SLr \tag{4.4}$$

where ALr equates the authorization level of resource k, ELr equates the entity protection level of resource r and SLr equate the security level of resource r.

5. **Capability (CA):** The term CA of the cloud resources gives a valuable performance of data or file transfer and application performance. It is calculated through CPU utilization parameters such as processor speed of resource r (Pr) and memory speed of resource r (Mr) and network parameters such as bandwidth(Br) and latency(Lr) of resource r.The CA of resources r is estimated via eq(4.5).

$$CAr = (Br * (Pr + Mr + (1/Lr)))/(Br * (PrMax + MrMax + (1/Lrmin))) \tag{4.5}$$

Br equate the quantity of data moved during the duration of the rth resource, Lr equate the delay to reach rth resource, MrMax equate the maximum speed of memory that exists in the system, Lrmin equate the minimum delay that exists in any link of the system and PrMax equate the maximum speed of processes that exist in the system.

6. **Trust Evaluation(TE):** In the earlier section, AVLr, RELr, DIr, IDr and CAr are found via the relationships presented. Using these values, we can find the value of TEr. The TE of resources r is estimated via eq(4.6).

$$TEr = ARDr + IDr + CAr \tag{4.6}$$

Where ARDr represent the trust value of resource(r) based on the AVLr, RELr and DIr.

4.3 Fuzzy logic

4.3.1 Overview

Fuzzy logic knowledge is like to the human being's sensation and interpretation process. Dissimilar standard control approach, which is a point-to-point controller, fuzzy logic control is a range-to-range or range-to-point control. The output of a fuzzy control is consequent from fuzzifications of together inputs and outputs by the related membership functions. A crisp input will be changed to the different members of the related membership functions created on its value. From this point of opinion, the output of a fuzzy logic control is created on its memberships of the dissimilar membership functions, which can be measured as an array of inputs.[21]

To implement fuzzy logic method to an actual application needs the subsequent three stages:

1. **Fuzzification:** Change crisp data or standard data into fuzzy data or Membership Functions (MFs)

2. **Fuzzy Inference Process:** Association membership functions by the control rules to arise the fuzzy output

3. **Defuzzification:** Usage dissimilar approaches to compute each related output and placed them into a table: the lookup table. Choice the output from the lookup table created on the existing input through an application

4.3.2 IF-THEN Rules

IF-THEN rules plan input or calculated truth values to wanted output truth values. Example:
IF temperature IS very cold THEN fan_speed is stopped
IF temperature IS cold THEN fan_speed is slow
IF temperature IS warm THEN fan_speed is moderate
IF temperature IS hot THEN fan_speed is high

Assumed a positive temperature, the fuzzy variable hot has a convinced truth value, which is derivative to the high variable. Must an output variable happen in numerous

THEN parts, before the values from the particular IF parts are collective by the OR operator.

4.3.3 Defuzzification

The goal is to become a constant variable from fuzzy truth values. This would be relaxed if the output truth values were precisely those gained from fuzzification of a certain number. Then, however, all output truth values are calculated independently, in most cases they do not characterize such a set of numbers. One has then to choose for a number that ties best the "intention" programmed in the truth value. For example, for numerous truth values of fan_speed, a real speed must be creating that best fits the calculated truth values of the variables 'slow', 'medium' and so on There is no single algorithm for this resolution. A common algorithm is:

- For separately truth value, cut the membership function at this value
- Association the subsequent curves using the OR operator
- Find the center-of-weight of the area below the curve
- The x location of this midpoint is then the final output

4.3.4 Fuzzy Interface System

Fuzzy Inference System is the main component of a fuzzy logic system taking decision making as its key work. It usages the "IF... THEN" rules along with connectors "OR" or "AND" for diagram critical decision rules.

- **Features of Fuzzy Inference System**
  Following are some features of FIS:
  - The output from FIS is continuously a fuzzy set regardless of its input which can be fuzzy or crisp.
  - It is essential to take fuzzy output when it is used as a controller.
  - A defuzzification component would be there with FIS to change fuzzy variables into crisp variables.

- **Functional Blocks of Fuzzy Inference System**

  The subsequent five functional blocks will benefit you recognize the structure of FIS:

  1. **Rule Base:** It covers fuzzy IF-THEN rules.
  2. **Database:** It describes the membership functions of fuzzy sets used in fuzzy rules.
  3. **Decision-making Component:** It achieves process on rules.
  4. **Fuzzification Interface Component:** It changes the crisp amounts into fuzzy amounts.
  5. **Defuzzification Interface Component:** It changes the fuzzy amounts into crisp amounts.

  Following figure 4.3 is a block diagram of fuzzy interference system.

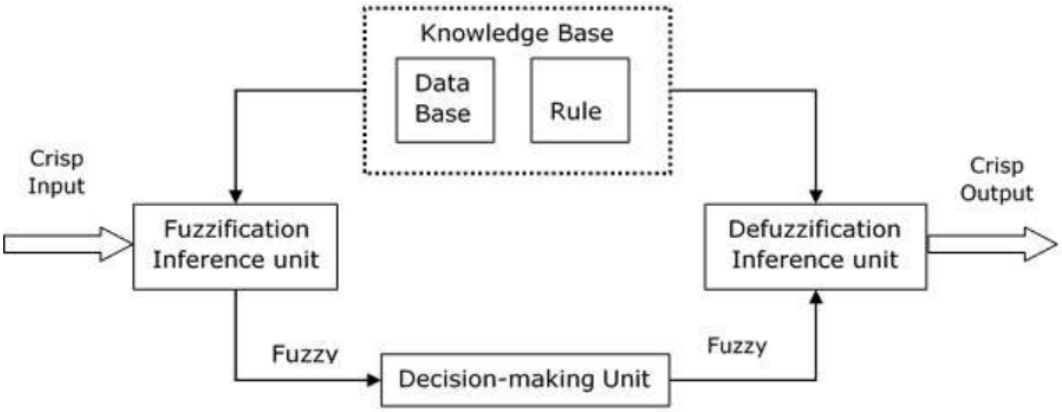


Figure 4.3: Block diagram of fuzzy interference system [21]

- **Working of Fuzzy Inference System**

  The working of the FIS contains of the following steps:

  – A fuzzification component provisions the application of many fuzzification methods and changes the crisp input into fuzzy input.

  – A knowledge base - assembly of rule base and record are formed upon the adaptation of crisp input into fuzzy input.

  – The defuzzification unit fuzzy input is to conclude changed into crisp output.

- **Methods of Fuzzy Interface System**

  Now discuss the dissimilar methods of FIS. Following are the two significant methods of FIS, having dissimilar consequential of fuzzy rules:

  - Mamdani Fuzzy Inference System
  - Takagi-Sugeno Fuzzy Model (TS Method)

4.3.5 Mamdani Fuzzy Inference System

This system was proposed in 1975 by Ebhasim Mamdani. Essentially, it was estimated to control a steam engine and boiler combination by manufacturing a set of fuzzy rules found from people working on the system.

- Steps for Computing the Output Following steps essential to be followed to calculate the output from this FIS:

  1. **Step 1:** Set of fuzzy rules want to be determined in this step.
  2. **Step 2:** In this step, via using input membership function, the input would be made fuzzy.
  3. **Step 3:** Now create the rule asset by merging the fuzzified inputs giving to fuzzy rules.
  4. **Step 4:** In this step, control the resultant of rule by merging the rule asset and the output membership function.
  5. **Step 5:** For receiving output circulation combine all the consequents.
  6. **Step 6:** Lastly, a defuzzified output distribution is gained.

  Following figure 4.4 is a block diagram of Mamdani Fuzzy Interface System.

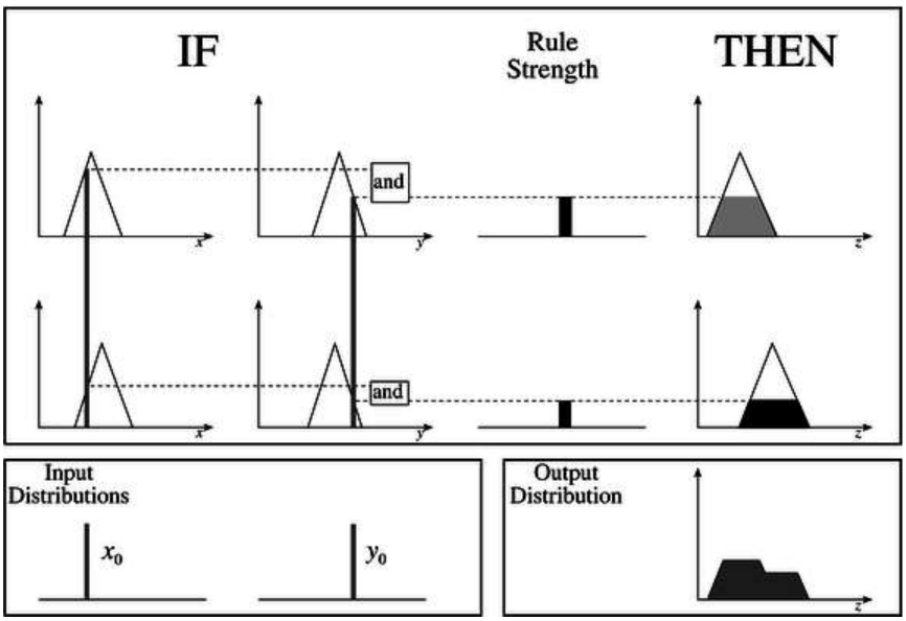


Figure 4.4: Block diagram of Mamdani Fuzzy Interface System [21]

4.4 Proposed Algorithm

Algorithm 1 Calculate Trust

1: **procedure** COLLECT THE DATA FROM DATA CENTRE. A, N, C, D, AL, SL, EL, B, L, P, M. (where A = No. of accepted jobs, N = No. of submitted jobs, C = No. of completed jobs, D = No. of data integrity preserved jobs, AL = Authorization Level, SL = Security Level, EL = Entity Protection Level, B = Bandwidth, L = Latency, P = Processor Speed, M = Memory Speed.)
2: $ARD \leftarrow ((N*C)+(A*C)+(N*D))/N*A*C$ (where ARD = Availability + Reliability + Data Integrity)
3: $ID \leftarrow AL + SL + EL$ (where ID = Identity)
4: $CA \leftarrow (B * P + M + (1/L)) / (B * Max(P) + Max(M) + (1/Min(L)))$ (where CA = Capability)
5: $TE \leftarrow ARD+ID+CA$ (where TE = Trust Evalution)
6: *Fuzzy Rules* $\leftarrow$ *TE(Low,Medium,High)*
7: **if** $TE >$ *High* **then return** Highly_Trusted(DC)
8: **else**
9: **if** *Low* <=TE<= *High* **then return** Trusted(DC)
10: **else**
11: **if** $TE <$ *Low* **then return** Highly_Untrusted(DC)
12: **close;**
13: *Output* $\leftarrow$ *Highly_Trusted(DC), Trusted(DC), Highly_Untrusted(DC)*

Chapter 5

Implementation & Result

In last proposed solution chapter we discuss about proposed model and proposed algorithm. That model and algorithm implement in this chapter and discuss in detail. In next section discuss about which tool and technology are used. Then discuss about MATLAB tool and cloudsim with it's result.

5.1 Tools and Technology

- Programming Language: JAVA
- Library: jFuzzyLogic, Fuzzy Logic Designer
- IDE: cloudsim
- Software: Eclipse and MATLAB

5.2 System Configuration

- Processor : Intel core i5 – 4790 CPU @ 3.60 GHz
- RAM: 8 GB
- Graphics: Intel(R) HD Graphics 4600
- OS Type: 64 – bits
- Operating System: Windows 8.0

5.3 Dataset

The simulation dataset is found from a practiced where it is available on www.dataset12.expertCloud.ir.[11] Expert Cloud as an original class of cloud systems allows its objects to demand the ability, data and proficiency without any data about their location by using Internet structures and cloud computing ideas. One of the greatest significant services in Expert Cloud is to exploration trustworthy objects to advantage from their data and services. We removed dataset of the Expert cloud acquire 2000 resources with dissimilar characteristics such as number of jobs accepted (Ar), number of jobs submitted (Nr), number of jobs completed successfully (Cr), number of jobs data integrity preserved (Dr), authorization level (ALr), entity protection level (ELr), security level (SLr), processor speed (Pr), memory speed (Mr), the amount of data transferred at the time (bandwidth(Br)), delay reaching resource (latency(Lr). The dataset values are shown in Table 5.1 briefly.

Resources	Ar	Nr	Cr	Dr	ALr	ELr	SLr	Br	Pr	Mr	Lr
1	12	18	6	0	0.10	0.05	0.08	28	20	40	21
2	15	21	12	1	0.78	0.86	0.65	60	43	40	73
3	13	24	7	2	0.11	0.13	0.18	59	2	21	32
.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....
2000	12	23	6	0	0.74	0.75	0.85	79	69	60	38

Table 5.1: Dataset

5.4 Fuzzy logic implement in MATLAB

Using dataset which is shown in section 5.3 and proposed algorithm which is shown in last chapter implement it in MATLAB and generate the result which are shown and discuss below.

The Trust FIS is the last Fuzzy model which takes the output of the past three blocks and gives the Trust rating as output. Be that as it may, the fuzzy model picked here is Mamdani FIS, so the output is a crisp value i.e. Low, Medium, High. This has extensively many standards contrasted with the past qualities because of the expansion in number of part elements of contribution and also output. A sample of rules is listed below.

1. If (ARD is High) and (ID is High) and (CA is High) then (TE is High) (1)

2. If ARD is High) and (ID is Medium) and (CA is Medium) then (TE is Medium) (1)

3. If (ARD is High) and (ID is Medium) and (CA is Low) then (TE is Medium) (1)

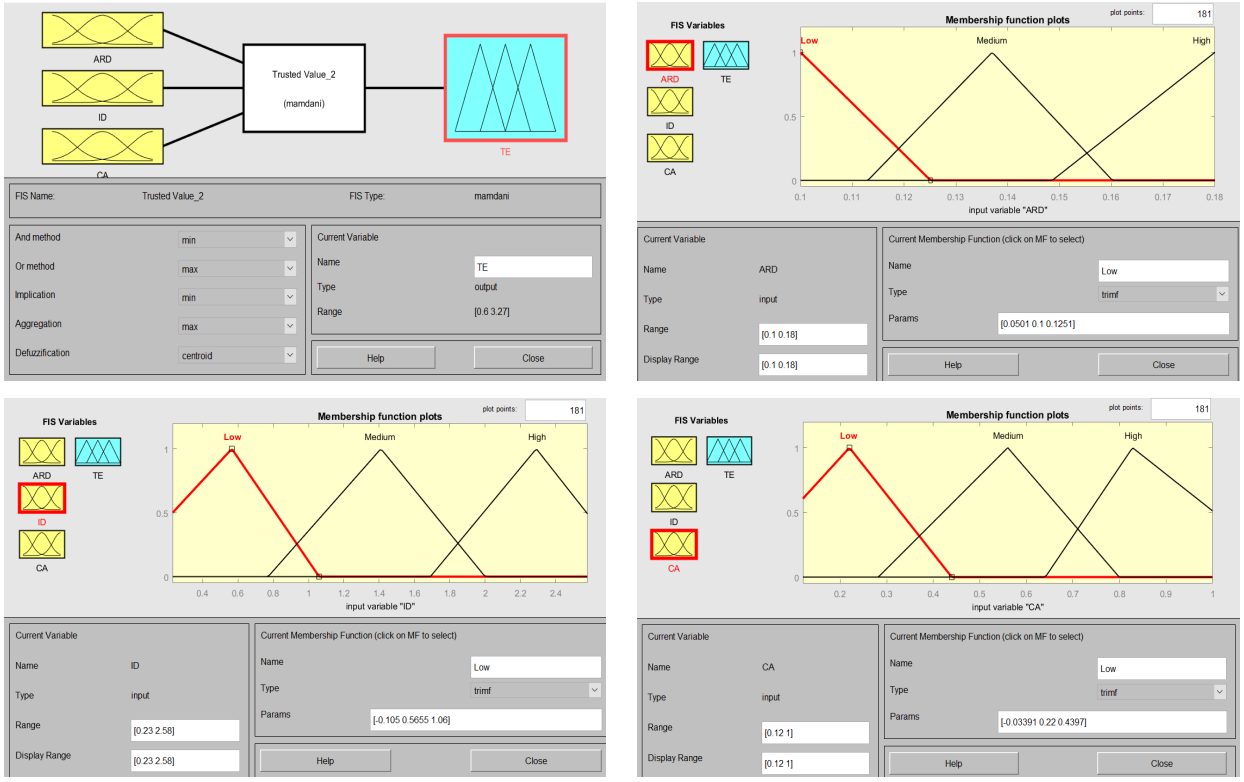
4. If (ARD is High) and (ID is Low) and (CA is High) then (TE is High) (1)

5. If (ID is Medium) then (TE is Medium) (1)

6. If (ID is Low) then (TE is Low) (1)

7. If (ARD is High) then (TE is High) (1)

Performance of FIS file is shown in three different tables with Figures. Figure 5.2 shown rules of fuzzy logic for trusted value. In table II shown different membership function for inputs and output. In table III shown different figures of surfaces with respect to ARD, ID, CA inputs and TE output. In table IV shown code of FIS.



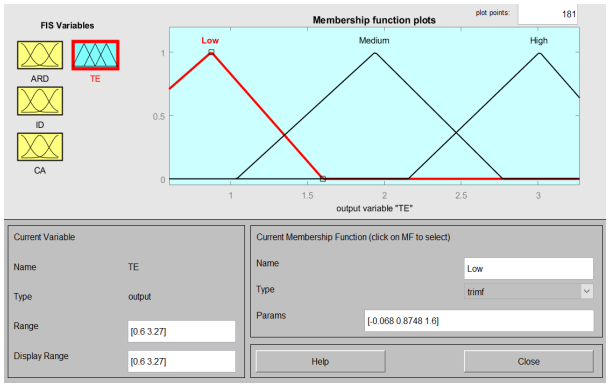


Table 5.2: Membership functions

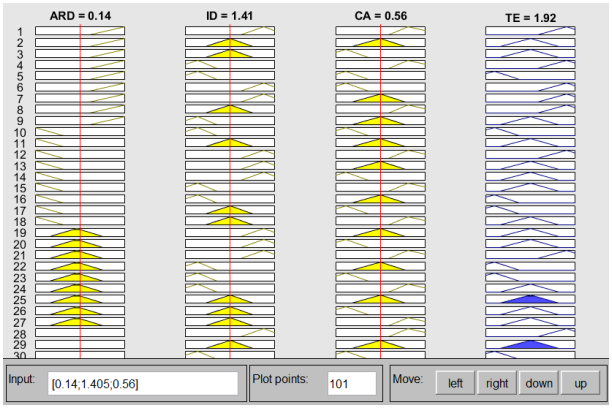
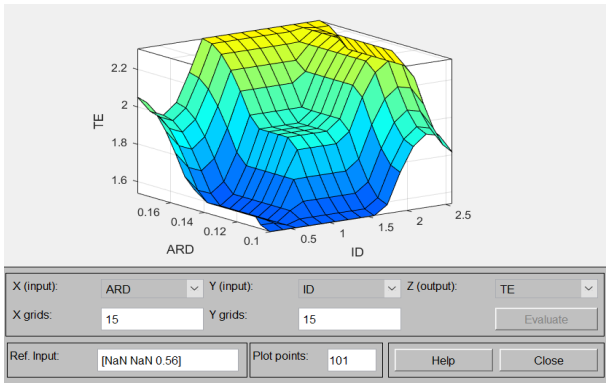
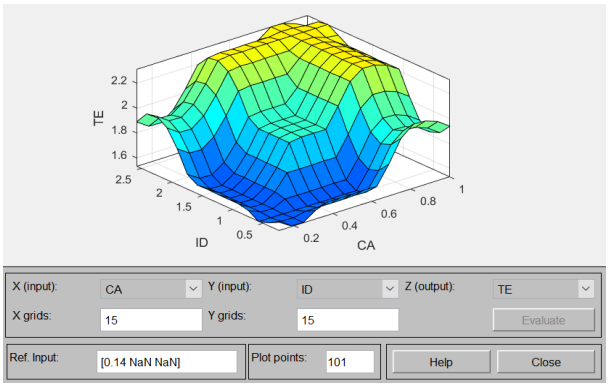
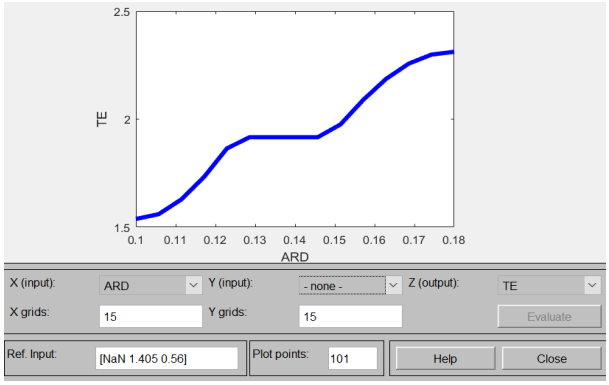
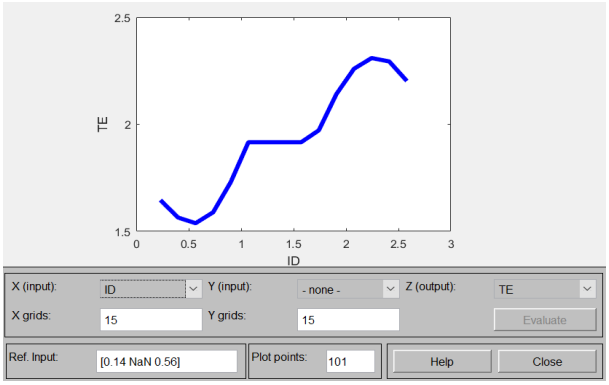


Figure 5.1: Rules









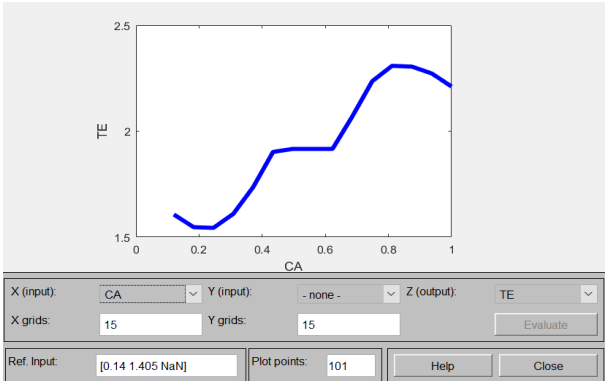
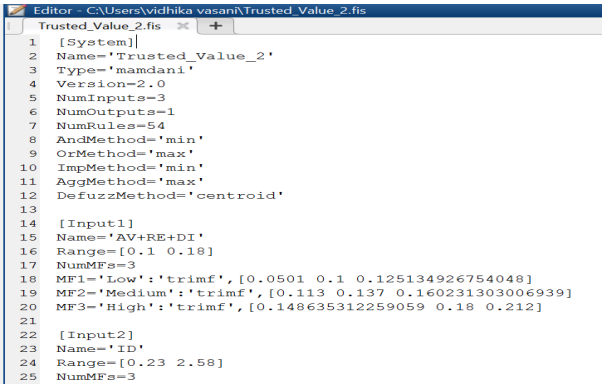


Table 5.3: Surfaces



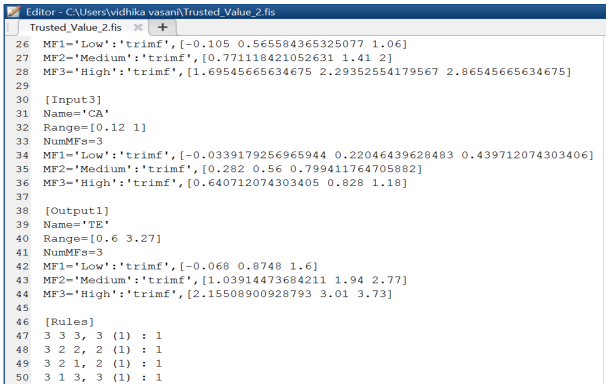


Table 5.4: FIS file in workspace

Input	Range of value	Member Functions	Output	Range of value	Member Functions
ARD	[0.0501,0.1000,0.1251]	Low	TE	[-0.0680,0.8748,1.6000]	Low
	[0.1130,0.1370,0.1602]	Medium			
	[0.1486,0.1800,0.2120]	High			
ID	[-0.1050,0.5655,1.0600]	Low		[1.0391,1.9400,2.7700]	Medium
	[0.7711,1.4100,2.0000]	Medium			
	[1.6954,2.2935,2.8654]	High			
CA	[0.0339,0.2204,0.4397]	Low		[2.1550,3.0100,3.7300]	High
	[0.2820,0.5600,0.7994]	Medium			
	[0.6407,0.8208,1.1800]	High			

Table 5.5: Result of MATLAB

5.5 Fuzzy logic implement in cloudsim

In this section implement fuzzy logic in cloudsim. Cloudsim is an open-simulator for colud and it is using eclipse platform.For fuzzy logic we have used jFuzzylogic jar file.[22] This effort presents jFuzzyLogic, an open source library for fuzzy structures which permit us to proposal Fuzzy Logic Supervisors secondary the ordinary for Fuzzy Control Software design available by the International Electrotechnical Commission. This library is

inscribed in Java and is accessible as open source from jfuzzylogic.sourceforge.net.Fuzzy Control Language is an industry ordinary requirement unrestricted by the International Electrotechnical Commission (IEC) as part of the Programmable Controller Languages (PLC) defined in the IEC-61131 requirement. In jFuzzyLogic FCL file is available. Using FCL generate same graph as like MATLAB and that jFuzzyLogic integrate into cloudsim and generate plot. In figure 5.2 shown membership function as poor, good and excellent with input varaible ARD, ID and TE with respect to resources of datacenter. In figure 5.3 shown result of jFuzzyLogic for ARD,ID, and TE as poor, good and excellent with respect to resources of datacenter(x).

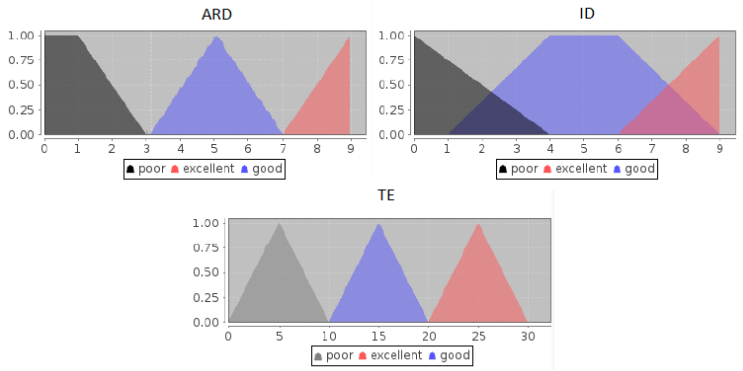


Figure 5.2: Membership Function

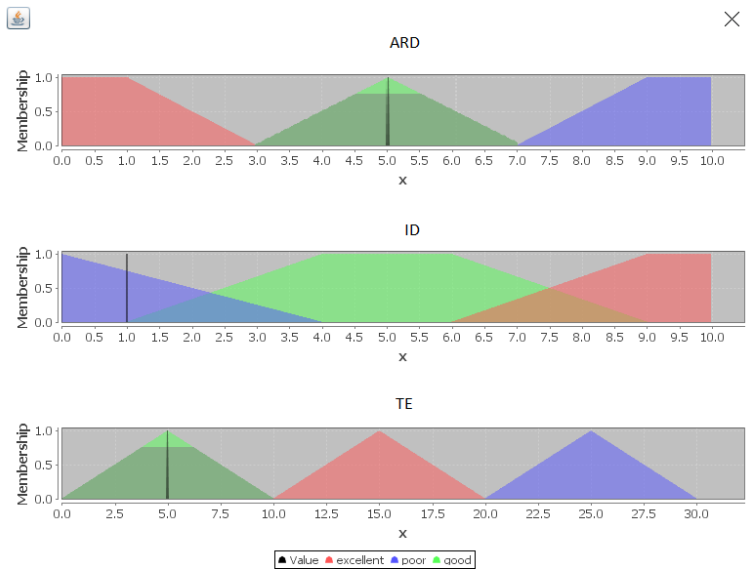


Figure 5.3: Result of jFuzzyLogic

5.6 Result Analysis

Here, shown the result of FIS file which has three different output for Highly trusted resources of datacenter, Trusted resources of datacenter and Highly untrusted resources of datacenter. It may help for CSP to select the resources for the datacenter. In figure 5.4 shown highly trusted resources of datacenter. In figure 5.5 shown trusted resources of datacenter. In figure 5.6 shown highly untrusted resources of datacenter.

	ard	capbility	ID	Trust
1	0.5325540	0.4133003	2.2903	3.2358543
3	0.6560590	0.5277778	2.0062	3.2220068
4	0.605988	0.477668	2.5813	3.664956
5	0.652597	0.691613	1.4064	2.750611
6	0.4899418	0.547317	1.7180	2.754735
9	0.506280	0.368221	1.7724	2.646901
10	0.495238	0.671776	1.5077	2.674714
11	0.587542	0.875676	2.1143	3.577518
12	0.593651	0.487734	1.1177	2.199085
13	0.609190	0.915716	1.2474	2.772305
15	0.568125	0.751443	0.9565	2.276068
16	0.630159	0.467662	2.2806	3.378420
17	0.611573	0.169234	2.7784	3.540207
18	0.655280	0.621965	1.5983	2.875544
19	0.650376	0.636890	2.0670	3.354266
20	0.621410	0.114500	2.3642	3.1001110
22	0.396135	0.403234	1.2706	2.269969
23	0.394831	0.457784	1.6791	2.531714
25	0.553380	0.930419	2.2109	3.694699
26	0.4960993	0.5970070	1.4866	2.579763
27	0.475135	0.860881	2.2836	3.619616
28	0.516162	0.309168	1.3799	2.205230
29	0.548583	0.671797	1.6685	2.888880
31	0.5108870	0.223980	2.1335	2.868350
33	0.492870	0.736611	1.6473	2.876781
34	0.611111	0.512622	2.6421	3.765833
35	0.626653	0.417983	2.0156	3.060236
36	0.679365	0.552299	1.9302	3.161865
38	0.427778	0.432922	1.8057	2.666399
40	0.508529	0.343389	1.4395	2.291419
...	...	...	...	...
1961	0.5497796	0.517551	1.9383	3.005547
1962	0.408333	0.488391	1.5181	2.414825
1963	0.689337	0.363238	1.4615	2.514075
1964	0.616809	0.497583	1.7012	2.815592
1965	0.420128	0.507576	1.6694	2.597104

Figure 5.4: Highly trusted resources of datacenter

	ard	capbility	ID	Trust
5	0.571429	0.497631	0.9788	2.047859
6	0.543810	0.512498	1.0340	2.090308
12	0.625977	0.219014	0.8263	1.671290
19	0.588333	0.209013	0.8437	1.641046
27	0.672365	0.940361	0.3475	1.960226
29	0.478066	0.079715	1.1171	1.674881
34	0.531328	0.403055	0.8068	1.741183
36	0.444444	0.582139	0.9717	1.998284
41	0.604469	0.233952	1.1570	1.995421
43	0.668981	0.442868	0.8481	1.959949
44	0.540424	0.442848	0.9731	1.956373
47	0.420228	0.432919	1.2833	2.136447
50	0.703846	0.547334	0.5826	1.833780
54	0.491111	0.283808	1.0420	1.816919
56	0.635522	0.159359	1.0219	1.816781
69	0.512169	0.437957	0.7342	1.684327
70	0.540217	0.552423	0.5293	1.621940
77	0.554155	0.358408	1.1730	2.085563
81	0.550414	0.134409	1.1151	1.799923
89	0.602315	0.069803	1.4400	2.112117
93	0.467995	0.388133	0.9468	1.802928
94	0.555051	0.190050	1.2963	2.041400
96	0.496093	0.472736	1.0904	2.059229
97	0.506280	0.781272	0.7648	2.052352
98	0.673913	0.507581	0.7204	1.901894
103	0.599206	0.373687	0.7604	1.733293
104	0.456078	0.801271	0.7330	1.990350
105	0.497987	0.226368	1.3718	2.096155
113	0.616088	0.597071	0.7608	1.973959
114	0.459596	0.588060	0.9095	1.957156
...	...	...	...	...
1757	0.434641	0.721619	0.6051	1.761360
1759	0.537698	0.373184	1.1586	2.069482

Figure 5.5: Trusted resources of datacenter

	ard	capbility	ID	Trust
0	0.388889	0.298744	0.2280	0.915633
2	0.455281	0.114583	0.4274	0.997264
24	0.671429	0.4612026	0.1909	1.474355
53	0.577749	0.328584	0.4186	1.304933
81	0.512169	0.875813	0.1491	1.537083
115	0.455556	0.527414	0.4572	1.440170
116	0.556624	0.477713	0.3580	1.392337
163	0.565920	0.049867	0.9391	1.554887
179	0.504233	0.432887	0.6522	1.589319
184	0.511753	0.228914	0.7843	1.524967
237	0.411111	0.532434	0.4936	1.437745
244	0.631766	0.224192	0.7221	1.578058
257	0.570707	0.467752	0.3093	1.347759
279	0.632143	0.258772	0.5959	1.486815
308	0.509868	0.442852	0.5192	1.471921
342	0.453363	0.502902	0.6366	1.592865
356	0.643551	0.3831135	0.3191	1.345786
391	0.412698	0.393129	0.2953	1.101127
416	0.548889	0.388127	0.6302	1.567216
423	0.395169	0.094609	1.0812	1.570978
438	0.576211	0.2886639	0.3780	1.242850
458	0.544974	0.278660	0.3273	1.150093
497	0.412582	0.114507	1.0708	1.597889
567	0.382222	0.373215	0.5894	1.344837
576	0.628594	0.358274	0.5392	1.526068
602	0.679293	0.248891	0.6006	1.526784
623	0.654971	0.303570	0.2739	1.232441
656	0.572222	0.273708	0.5456	1.391531
706	0.546296	0.279602	0.5981	1.423998
738	0.472955	0.379104	0.7133	1.563860
...	...	...	...	...
1182	0.639434	0.263785	0.3997	1.302919
1220	0.408730	0.054801	0.8524	1.315931
1264	0.633333	0.562285	0.3541	1.549718
1373	0.409018	0.144374	0.9579	1.511292
1397	0.659117	0.462757	0.2954	1.417274

Figure 5.6: Highly untrusted resources of datacenter

Now discuss about result of rated datacenter which is generated using jfuzzyLogic jar file which shown into figure 5.7, 5.8 and 5.9 respect to High rate trusted resources of datacenter, Medium rate trusted resources of datacenter and Low rate trusted resources of datacenter.

In the end of this chapter shown the comparison graph of proposed algorithm with respect to Highly trusted resources of datacenter, Trusted resources of datacenter and Highly untrusted resources of datacenter. Comparison Graph of proposed algorithm is shown into figure 5.10.

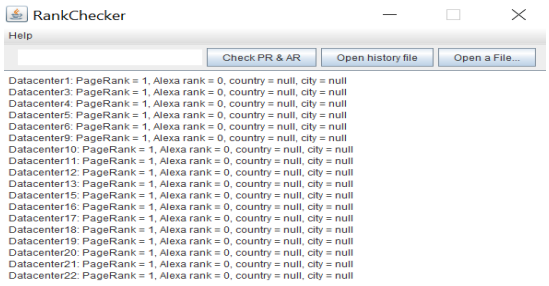


Figure 5.7: High rate trusted resources of datacenter

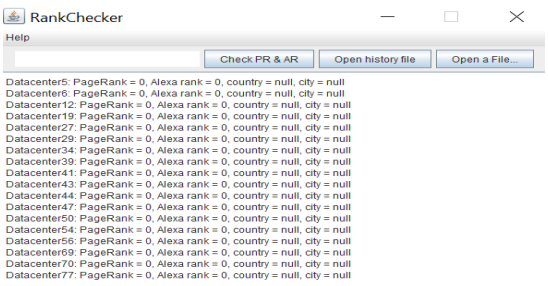


Figure 5.8: Medium rate trusted resources of datacenter

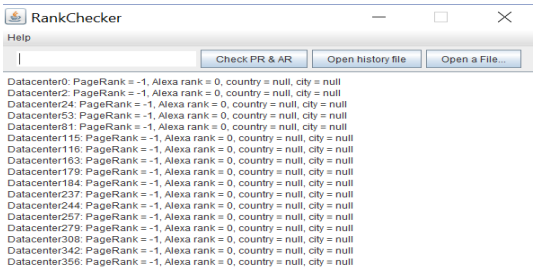


Figure 5.9: Low rate trusted resources of datacenter

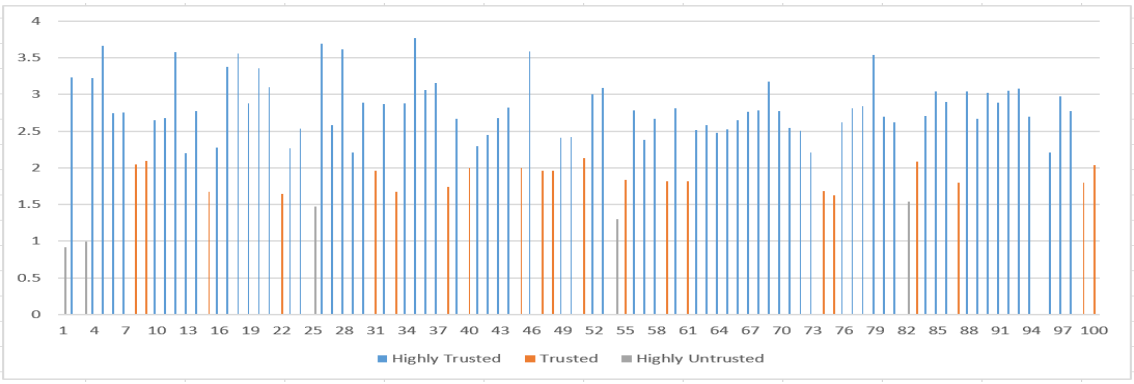


Figure 5.10: Comparison of resources of datacenter

Chapter 6

Conclusion & Future Work

6.1 Conclusion

We have classified the various existing techniques of trusted cloud computing We have examined effects of fuzzy logic performed with various parameters: number of jobs accepted (Ar), number of jobs submitted(Nr), number of jobs completed successfully (Cr), number of jobs data integrity pre-served (Dr), authorization level (ALr), entity protection level (ELr), security level (SLr),processor speed (Pr), memory speed (Mr), the amount of data transferred at the time(bandwidth(Br)), delay reaching resource (latency(Lr). Further, we computed trust via characteristics of resources like Availability, Reliability, Data integrity, Capability and Identity. The results show the Highly trusted resources of the datacenter, Trusted resources of datacenter and Highly untrusted resources of datacenter using MATLAB tool and cloudsim. At the end of this thesis, we experiment FIS file into the cloudsim using jFuzzyLogic and generate the result as like as MATLAB tool. Through cloudsim CSP can provide high trusted resources of the datacenter to the client. CSP can serve Trust as a Service(TaaS) using this proposed algorithm to the client.

6.2 Future Work

In future, we can use the machine learning classification techniques: K-means clustering, C-means clustering etc. We can explore more Trust as a Service(TaaS) on datacenter and Hypervisors. Also, we can work on policy-based trust and prediction based trust. We can also use different techniques to secure the giving rated resources of the datacenter.

Bibliography

[1] A. Singh and K. Chatterjee, "Cloud security issues and challenges: A survey," *Journal of Network and Computer Applications*, vol. 79, pp. 88–115, 2017.

[2] R. K. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, and B. S. Lee, "Trustcloud: A framework for accountability and trust in cloud computing," in *Services (SERVICES), 2011 IEEE World Congress on*, pp. 584–588, IEEE, 2011.

[3] M. B. Monir, M. H. AbdelAziz, A. A. AbdelHamid, and E.-S. M. EI-Horbaty, "Trust management in cloud computing: A survey," in *Intelligent Computing and Information Systems (ICICIS), 2015 IEEE Seventh International Conference on*, pp. 231–242, IEEE, 2015.

[4] K. Hwang and D. Li, "Trusted cloud computing with secure resources and data coloring," *IEEE Internet Computing*, vol. 14, no. 5, pp. 14–22, 2010.

[5] J. Huang and D. M. Nicol, "Trust mechanisms for cloud computing," *Journal of Cloud Computing: Advances, Systems and Applications*, vol. 2, no. 1, p. 9, 2013.

[6]

[7] T. H. Noor, Q. Z. Sheng, L. Yao, S. Dustdar, and A. H. Ngu, "Cloudarmor: Supporting reputation-based trust management for cloud services," *IEEE transactions on parallel and distributed systems*, vol. 27, no. 2, pp. 367–380, 2016.

[8] M.-J. Sule, M. Li, and G. Taylor, "Trust modeling in cloud computing," in *Service-Oriented System Engineering (SOSE), 2016 IEEE Symposium on*, pp. 60–65, IEEE, 2016.

[9] P. Sen, P. Saha, and S. Khatua, "A distributed approach towards trusted cloud computing platform," in *Applications and Innovations in Mobile Computing (AIMoC), 2015*, pp. 146–151, IEEE, 2015.

[10] M. Firdhous, O. Ghazali, and S. Hassan, "A trust computing mechanism for cloud computing," in *Kaleidoscope 2011: The Fully Networked Human?-Innovations for Future Networks and Services (K-2011), Proceedings of ITU*, pp. 1–7, IEEE, 2011.

[11] M. Chiregi and N. J. 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*, vol. 60, pp. 280–292, 2016.

[12] T. Grandison and M. Sloman, "Trust management tools for internet applications," in *International Conference on Trust Management*, pp. 91–107, Springer, 2003.

[13] M. Supriya *et al.*, "Estimating trust value for cloud service providers using fuzzy logic," 2012.

[14] Y. K. Tamandani and M. U. Bokhari, "Sepfl routing protocol based on fuzzy logic control to extend the lifetime and throughput of the wireless sensor network," *Wireless networks*, vol. 22, no. 2, pp. 647–653, 2016.

[15] S. Jain *et al.*, "A trust model in cloud computing based on fuzzy logic," in *Recent Trends in Electronics, Information & Communication Technology (RTEICT), IEEE International Conference on*, pp. 47–52, IEEE, 2016.

[16] M. A. H. Monil and R. M. Rahman, "Vm consolidation approach based on heuristics, fuzzy logic, and migration control," *Journal of Cloud Computing*, vol. 5, no. 1, p. 8, 2016.

[17] P. Pandey and S. Singh, "Fuzzy logic based job scheduling algorithm in cloud environment,"

[18] A. N. Toosi and R. Buyya, "A fuzzy logic-based controller for cost and energy efficient load balancing in geo-distributed data centers," in *Utility and Cloud Computing (UCC), 2015 IEEE/ACM 8th International Conference on*, pp. 186–194, IEEE, 2015.

[19] A. Patil and H. Chaudhari, "Modeling fuzzy scheduling in infrastructure as a service cloud," *International Journal of Computer Applications*, vol. 98, no. 13, 2014.

[20] U. M. A. Naushahi, "Profile-based access control in cloud computing environments with applications in health care systems," 2016.

[21] Y. Bai and D. Wang, "Fundamentals of fuzzy logic control—fuzzy sets, fuzzy rules and defuzzifications," in *Advanced Fuzzy Logic Technologies in Industrial Applications*, pp. 17–36, Springer, 2006.

[22] P. Cingolani and J. Alcala-Fdez, "jfuzzylogic: a robust and flexible fuzzy-logic inference system language implementation," in *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on*, pp. 1–8, IEEE, 2012.