B.M.S College of Engineering

P.O. Box No.: 1908 Bull Temple Road,
Bangalore-560 019

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Course – Cloud Computing
Course Code –20IS5PCCLC
AY 2021-2022

Cloud Architecture

Submitted to – Anitha H M
(Assistant Professor)

Submitted by -
Niranjan Hegde 1BM19IS103
Prashanth Jaganathan 1BM19IS115
Prateek Gummaraju 1BM19IS117
Samartha S 1BM19IS219

B.M.S College of Engineering

P.O. Box No.: 1908 Bull Temple Road,
Bangalore-560 019

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CONTENTS

ABSTRACT

Cloud computing is the on-demand availability of computer system resources, especially data storage and computing power, without direct active management by the user. It is a computing environment where computing needs by one party can be outsourced to another party and when need arises to use the computing power or resources like database or emails, they can access them via the internet. Cloud computing architecture basically consists of Front End (Client Infrastructure) and Back End (Application, Service, Security, Storage, etc). They both are connected to each other generally through the internet. In the Literature Survey, we will be reviewing the existing cloud architecture, the proposed and upcoming architectures and the various applications of these architectures. These applications include cloud architecture for mobile computing environments, smart homes, remote sensors (IoT), predictive analysis (Machine Learning), reducing on-demand gaming latency etc.

CHAPTER 1

1.1) Introduction

Cloud computing refers to the on-demand availability of computer system resources, particularly data storage and computational power, without the user having to manage them directly. It is a computer environment in which one party's computing needs can be outsourced to another, and when the need arises to use the computing power or resources, such as databases or emails, they can do so via the internet. Front End (Client Infrastructure) and Back End (Application, Service, Security, Storage, etc) are the two main components of cloud computing architecture. They are both connected to each other via the internet in general.

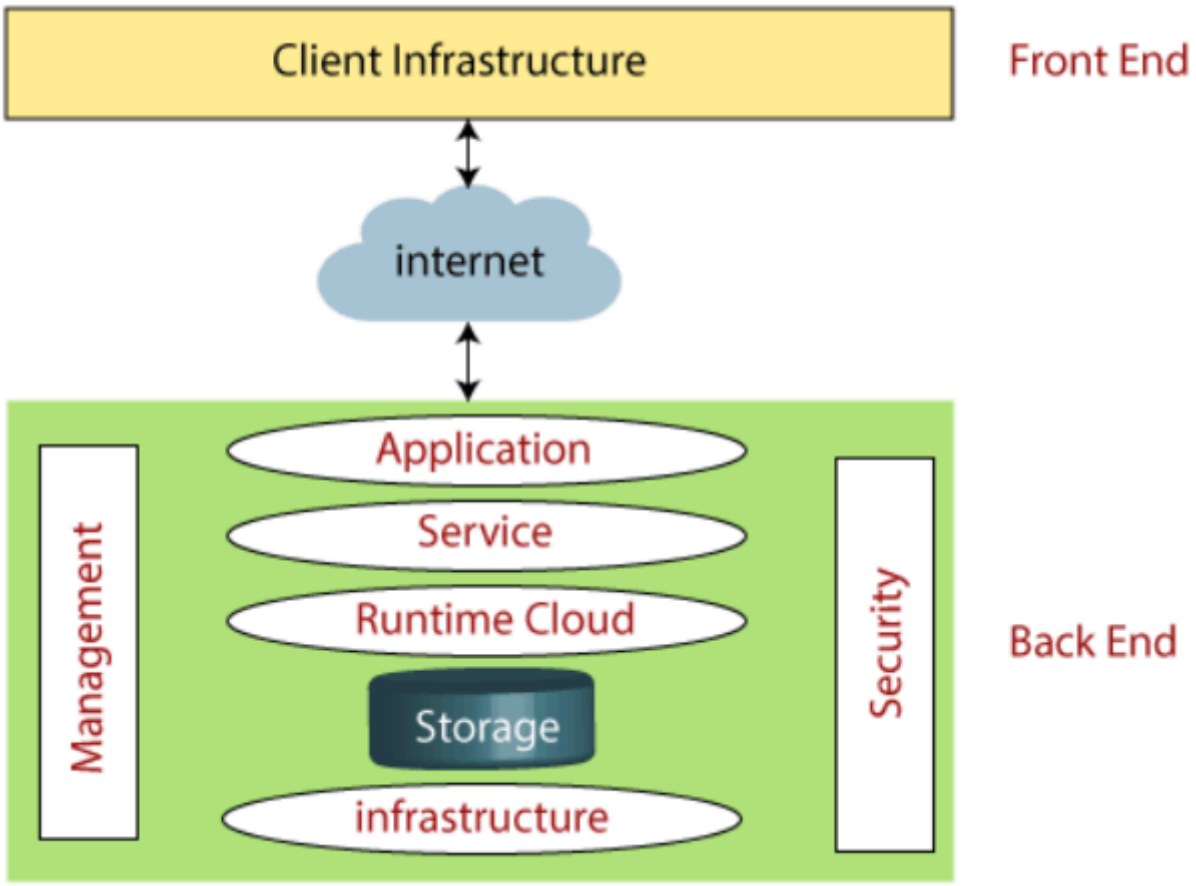

Fig 1. Basic Cloud Architecture

The 3 main layers of cloud computing architecture are:

a) Application/ Software
b) Platform
c) Infrastructure

a) Software as a Service: A cloud application delivers "Software as a Service (SaaS)" over the internet, thus eliminating the need to install and run the application on the users' system. Examples are Google Apps, Salesforce Dropbox, Slack, Hubspot, Cisco WebEx.
b) Platform as a Service: Platform services "Platform as a Service (PaaS)" provide a computing platform using the cloud infrastructure. It has all the applications typically required by the client deployed on it. Thus the client need not go through the hassles of buying and installing the software and hardware required for it. Examples are Windows Azure, Force.com, Magento Commerce Cloud, OpenShift.
c) Infrastructure as a Service: Infrastructure services "Infrastructure as a Service (IaaS)" provides the required infrastructure as a service. The client need not purchase the required servers, data center or the network resources. Also the key advantage here is that customers need to pay only for the time duration they use the service. Examples are Amazon Web Services (AWS) EC2, Google Compute Engine (GCE), Cisco Metapod.

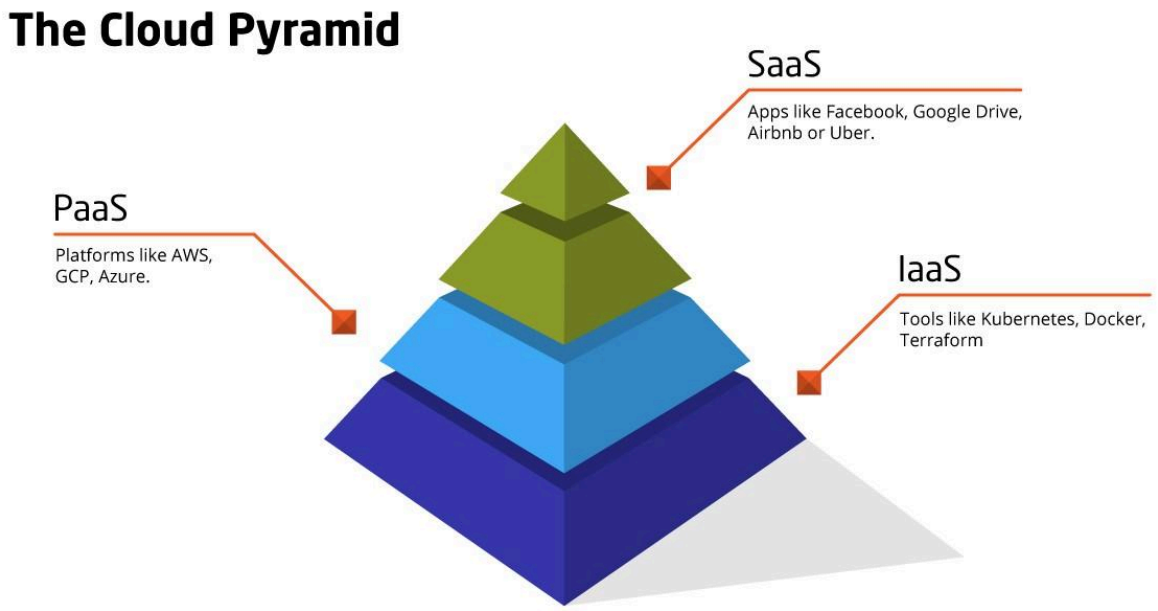

Fig 2. Layers of cloud computing architecture

Secure Cloud Architecture: The primary distinction between cloud computing and typical corporate internal IT services is that cloud IT infrastructures have independent owners and users. In cloud computing, this development necessitates a security duty split. Cloud service providers (CSP) must protect the services they supply and must not exceed the authority of their customers. The biggest challenge in cloud computing is the security and privacy problems caused by its multi-tenancy nature and the outsourcing of infrastructure, sensitive data and critical applications.

Mobile Cloud Computing: By exploiting cloud computing and transferring mobile workloads to the cloud for remote execution, the performance of mobile computing might be considerably improved. Edge cloud's hierarchical architecture allows for the aggregation of peak loads over several tiers of cloud servers, allowing for the maximum amount of mobile workloads to be handled. We also offer a workload placement method that determines which edge cloud servers mobile programmes are placed on and how much computational capability is supplied to execute each programme, in order to ensure efficient use of cloud resources.

Sensor-Cloud: Wireless sensor network (WSN) applications are now being employed in a variety

of fields, including healthcare, the military, critical infrastructure monitoring, environmental monitoring, and manufacturing. However, because of the constraints of WSNs in terms of memory, energy, compute, communication, and scalability, efficient data management in these areas is a critical issue to address. For real-time processing and storing of WSN data, as well as analysis (online and offline) of the processed information under context using inherently complicated models to extract events of interest, a powerful and scalable high-performance computing and huge storage infrastructure is required.

On-demand gaming architecture: Cloud gaming, also known as gaming on demand or gaming-as-a-service, is a form of internet gaming in which video games are hosted on remote servers and streamed directly to a user's device, or, more informally, playing a game from the cloud. It differs from traditional gaming, in which a user's video game console, personal computer, or mobile device executes the game locally. It is demonstrated through a large-scale measurement study that the existing cloud infrastructure is unable to meet the strict latency requirements necessary for acceptable on-demand game play. It is found that a hybrid infrastructure significantly improves the number of end-users served.

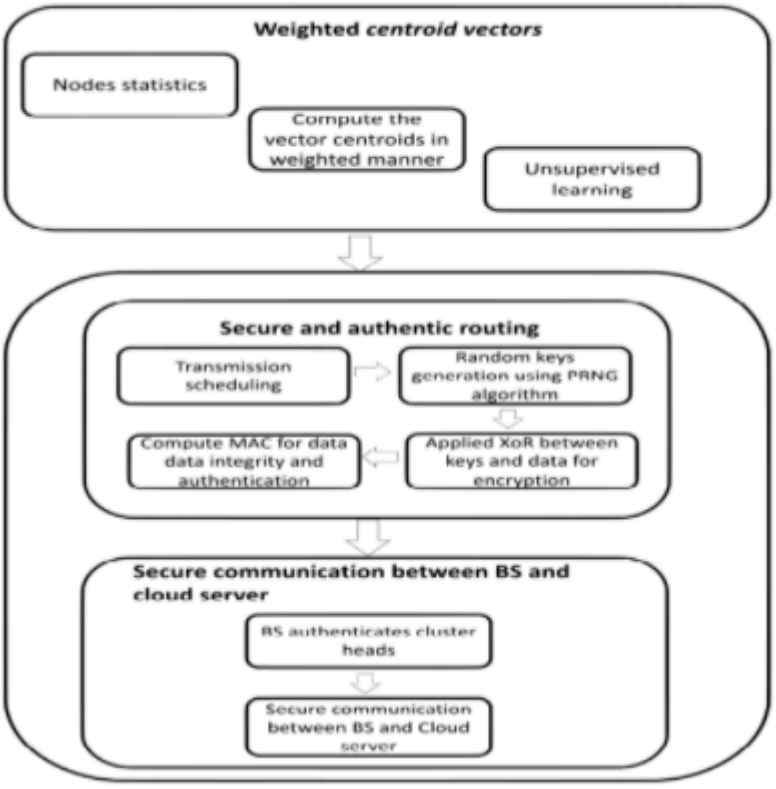

Fig. 3 Sensor Cloud Architecture

Cloud architecture in smart homes: All the current Cloud architecture implementations are computer-based, hence there is no exploitation on the computing power of digital home appliances and potential users of smart home. In smart home, the key electrical appliances and services, connected with each other, must be remotely controlled, monitored or accessed in order to form a communications network. Cloud computing and smart home are developing rapidly in their own fields, but currently no research work that combines them with each other has been done.

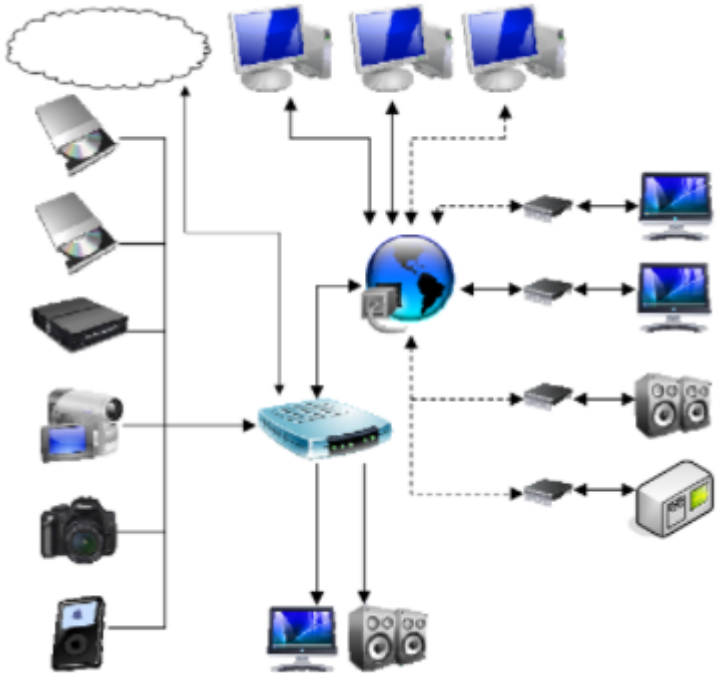

Fig. 4 Network structure of a smart home

CHAPTER 2

2.1) Literature survey

Trusted Computing Environment Model in Cloud Architecture:

Cloud computing is a new type of IT service in which customers make use of the cloud computing infrastructures such as CPU capability, network and storages provided by cloud service providers (CSP). Cloud computing can help to reduce IT cost of small and medium enterprises (SME) in that they need not to buy their own IT infrastructures and employ an IT team again. However, when it comes to transfer their businesses to cloud, people tend to worry about the privacy and security.

Is CSP trustworthy? Will the concentrated resources in cloud be more attractive to hackers? Will the customers be locked in to a particular CSP? All these concerns constitute the main obstacle to cloud computing. Cloud Security Alliance (CSA) describes these characteristics as: abstraction of infrastructure, resource democratization, services-oriented architecture, elasticity/dynamism of resources and utility model of consumption & allocation. NIST summarizes these characteristics as:

on-demand self-service, ubiquitous network access, resource pooling, rapid elasticity and pay per use. Since these cloud facilities are shared resources and generally located in the data centre of CSP, they are under the full control of CSP. Security devices in cloud are also owned and controlled by CSP. On the other hand, customers have no control over the facilities on which their businesses run.

This paper presents a multi-tenancy trusted computing environment model (MTCEM) to support the security duty separation between CSP and customers. MTCEM is designed for IaaS service delivery model, and it intends to separate the security responsibility of cloud infrastructures including hosts and virtual instances based on CSP and customers respectively. In MTCEM model, CSP is responsible to assure a trusted host and VMM environment and customers are responsible for the assurance of trusted virtual instances they rent from CSP. MTCEM is based on trusted computing platform technology advocated by Trusted Computing Group (TCG).

Trusted Computing Group (TCG) is a not-for-profit organization formed to develop, define and promote open, vendor-neutral, industry standards for trusted computing building blocks and software interfaces across multiple platforms. The main idea of TCG is to assure computing

platform trusted based on a hardware protected cryptograph module named Trusted Platform Module (TPM) and related software stacks. Another important mechanism of trusted computing platform technology is platform attestation. Attestation is a mechanism by which a computing platform proves to a third party that it is trusted.

Multi-tenancy trusted computing environment model (MTCEM) is a security duty separation model designed mainly for IaaS, and its purpose is to assure that CSP will secure the infrastructures they provide as services and that customers must build trusted virtual instances for themselves.

Both sides will not exceed to other's authorities. MTCEM is a two-level hierarchy transitive trust chain model which supports the security duty separation. In MTCEM, we have three entities as follow:

- Customers, who rent cloud services from CSP.
- CSP, who provide IaaS services.
- The Third-Party Auditor, optional but recommended, who is responsible to verify whether the infrastructures provided by CSP are trusted on behalf of customers.

MTCEM was initially designed for service delivery IaaS, however, it is also applied to PaaS service delivery model. In a PaaS environment, CSP may offer a JVM development environment to customers. Thus, CSP must assure the JVM platforms are trusted, and customers need to be responsible for the security of their own java applications. The challenge of this model is to implement a reasonable co-management mechanism of CSP and cloud customers on the cloud computing platforms owned by CSP. MTCEM makes its main contribution in that it presents a new mechanism of two-level hierarchy transitive trust chain to meet the requirement of cloud customers to control the computing platforms in cloud. Another new feature of MTCEM is that it leverages the platform attestation mechanism to make assure the IaaS platforms provided by CSP are trusted. On the other hand, MTCEM can also release CSP from the overwork of the cloud platform management.

Secure Cloud Computing:

With Cloud Computing becoming a popular term on the Information Technology (IT) market, security and accountability has become important issues to highlight. There are a number of security issues/concerns associated with cloud computing but these issues fall into two broad

categories: Security issues faced by cloud providers (organizations providing Software-, Platform-, or Infrastructure-as-a-Service via the cloud) and security issues faced by their customers. In most cases, the provider must ensure that their infrastructure is secure and that their clients' data and applications are protected while the customer must ensure that the provider has taken the proper security measures to protect their information.

External network attacks in the cloud are increasing at a notable rate. Malicious user outside the Cloud often performs DoS or DDoS attacks to affect the availability of Cloud services and resources. Port scanning, IP spoofing, DNS poisoning, phishing is also executed to gain access of Cloud resources. A malicious user can capture and analyse the data in the packets sent over this network by packet sniffing. IP spoofing occurs when a malicious user impersonates a legitimate users IP address where they could access information that they would not have been able to access otherwise. Internal attacker (authorized user) can easily get access to other user's resources without being detected. An insider has higher privileges and knowledge (related to network, security mechanism and resources to attack) than the external attacker. Therefore, it is easy for an insider to penetrate an attack than external attackers.

Vulnerabilities in Cloud can be defined as the loopholes in the security architecture of Cloud, which can be exploited by an adversary via sophisticated techniques to gain access to the network and other infrastructure resources. Some of the major Cloud specific vulnerabilities, which pose serious threats to Cloud Computing discussed in this paper are:

- Session Riding and Hijacking
- Reliability and Availability of Service
- Insecure Cryptography
- Data Protection and Portability

By exploiting vulnerabilities in Cloud, an adversary can launch the following attacks:

- Zombie attack
- Service injection attack
- Attacks on virtualization
- Rootkit in Hypervisor
- Man-in-the Middle attack

This paper proposes a cloud security architecture, which protect organizations against security threats and attacks. The key points for this architecture based on our analysis of existing security

technologies are:

- Single Sign-on (SSO): to streamline security management and to implement strong authentication within the cloud, organizations should implement Single Sign-On for cloud users. This enables user to access multiple applications and services in the cloud computing environment through a single login, thus enabling strong authentication at the user level.
- Defence in depth Security Approach: Virtual firewall appliances should be deployed instead of first-generation firewalls. This allows network administrators to inspect all levels of traffic, which includes basic web browser traffic, to peer-to-peer applications traffic and encrypted web traffic in the SSL tunnel. Intrusion-Prevention-Systems (IPS) should be installed to protect networks from internal threats from insiders.
- Increase Availability: This can be done by implementing high availability technologies such as active/active clustering, dynamic server load balanced and ISP load balancing within the network infrastructure.
- Data Privacy: Data loss prevention (DLP) tools can help control migration of data to the cloud and also find sensitive data leaked to the cloud. Data loss prevention (DLP) is a strategy for making sure that end users do not send sensitive or critical information outside of the corporate network. DLP help a network administrator control what data end users can transfer.
- Data Integrity: User can do the increase and decrease of the data capacity in the cloud server with the help of CSP(cloud service provider) in his request. This storage level must be with flexible and durability condition as far as its entire design or structure is concerned.
- Virtual Machine Protection: To adequately protect virtual machines, they should he isolated from o t h e r network segments and deep inspection at the network level should be implemented to prevent them both from internal and external threats. Illegal internal access should be restricted by implementing intrusion prevention systems and unauthorized external access should be protected by using secure remote access technologies like IPSec or SSL VPN.

In this paper, a physical cloud computing security architecture has been presented. In future, the proposed architecture may be modified with the advancement of security technologies used for implementing this physical cloud security architecture. By considering the contributions from several IT industries worldwide, it’s obvious that cloud computing will be one of the leading strategic and innovative technologies in the near future.

One of the most important challenges in mobile computing is to address the contradiction between the increasing complexity of mobile applications and the limited local capabilities of mobile devices. A viable solution to this challenge is to leverage cloud computing and execute mobile applications remotely. But the normal cloud architecture cannot be directly used for mobile computing. There are pressing needs to redesign the cloud architecture to serve the mobile workloads with better efficiency and scalability. To efficiently handle the peak load and satisfy the requirements of remote program execution, one can deploy cloud servers at the network edge and design the edge cloud as a tree hierarchy of geo-distributed servers, so as to efficiently utilize the cloud resources to serve the peak loads from mobile users. The hierarchical architecture of edge cloud enables aggregation of the peak loads across different tiers of cloud servers to maximize the amount of mobile workloads being served.

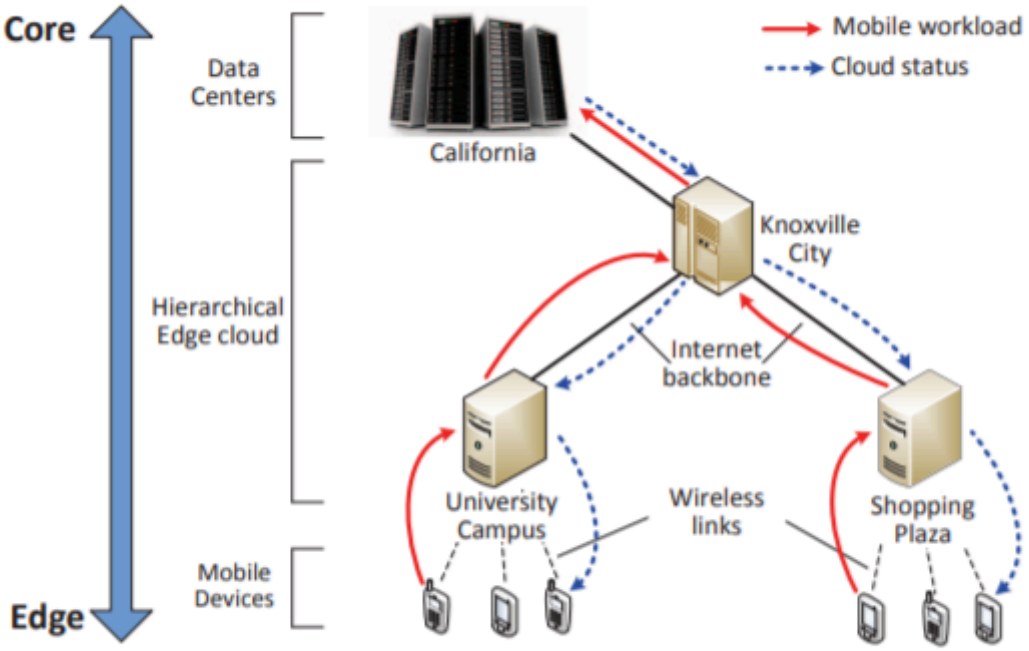

Fig. 1: The hierarchical edge cloud architecture

The Cloud Computing paradigm is one of the most disruptive technology advances of our times. This paradigm has made Information Technology (IT) resources available to the general public through the Internet. The progress of communication technologies has also contributed to this end. Boosting the bandwidth and hence speed of all connections has enabled us to handle more traffic.These cloud-based services evolutions have resulted in an increased outsourcing work to the cloud in all areas. This paradigm is now growing many times faster than the rest of the IT industry. Implementation of this paradigm to mobile computation has led to the Mobile Cloud Computing (MCC) concept. It consists in outsourcing part of the processing load to the Cloud and getting back the results. The implications for mobile devices are evident: in the first place, they can increase their performance without any change in their hardware, and, secondly, they allow us to extend the life of battery powered devices. These benefits have enhanced the potential

of the Internet ofTings (IoT) paradigm and allow us to compute advanced applications on devices and embedded systems. The main objective of the paper is researching architecture models that are able to increase the resilience and flexibility of offloading the workload outside the device and to enable scheduling the tasks along the network.

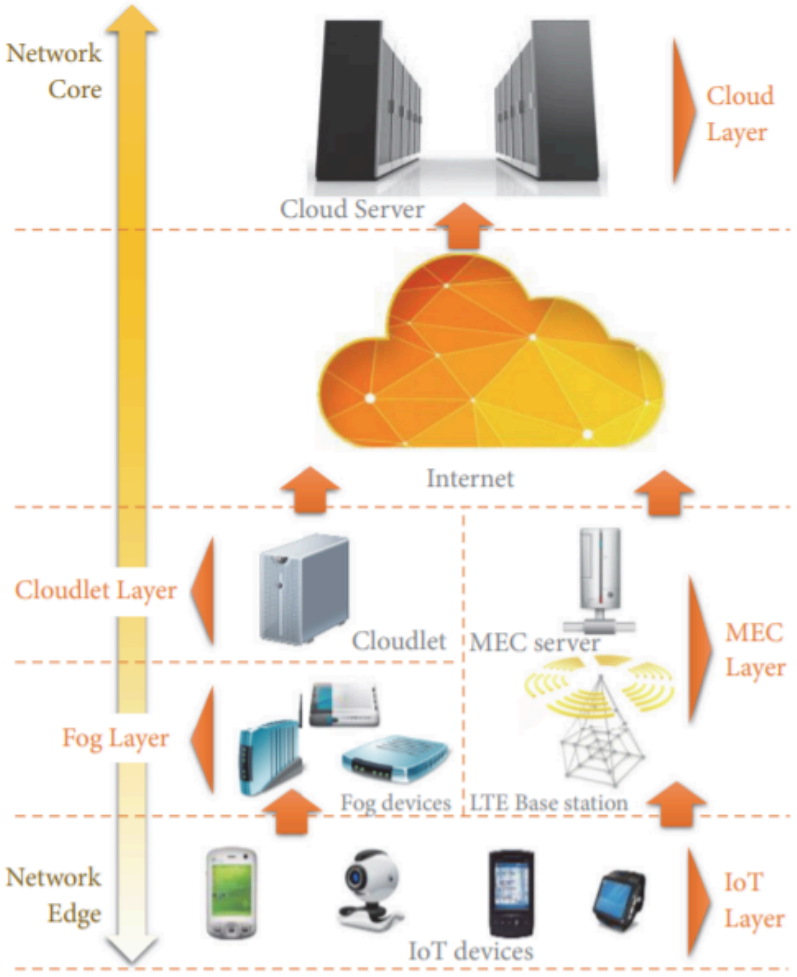

FIGURE 1: Outsourcing options deployed along the communication network.

Smart home refers to a smart space environment built at a digital home with pervasive computing algorithms in order to provide more humanized services. In a smart home, the key electrical appliances and services, connected with each other, must be remotely controlled, monitored or accessed in order to form a communications network. This research paper proposes a Cloud architecture which is based on digital appliances in smart homes to provide IT Services that can be accessed and used by appliances users. The link between the Cloud and smart home, is the home router which helps smart homes merge into Cloud to provide more information services and access services provided by Cloud, plays a significant role in this architecture. This Cloud architecture extends Cloud computing applications to a new domain so that more users could benefit from Clouds.

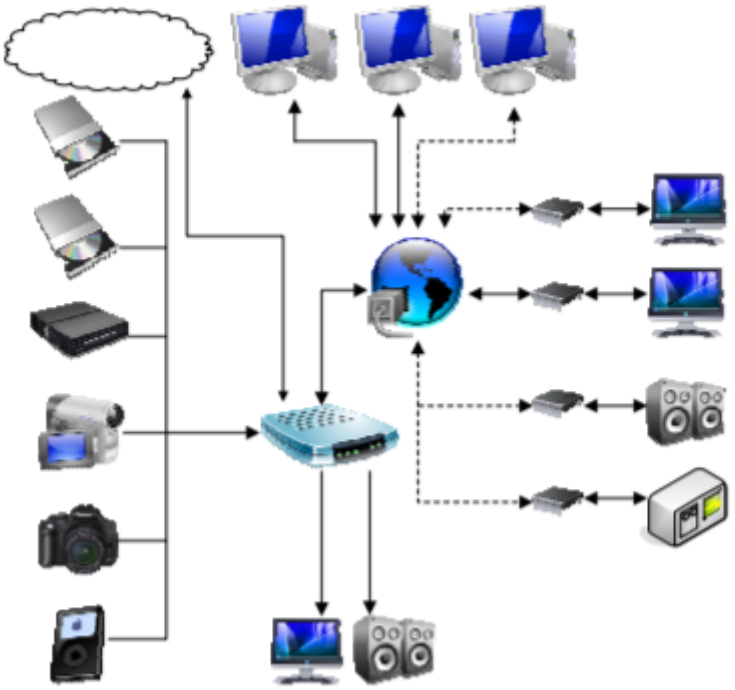

Figure 1. Network structure of smart home

Cloud Framework for Machine Learning approach:

This provides a distributed architecture to machine learning practitioners with a set of tools and cloud services that cover the whole machine learning development cycle: ranging from the models creation, training, validation and testing to the models serving as a service, sharing and publication.The DEEP framework is designed for allowing scientists to develop machine learning and deep learning models on distributed e-Infrastructures. It provides a comprehensive framework that covers the whole machine learning develop-ment cycle. For this different high level components have been designed, depicted in Fig 5.and described in what follows.

The DEEP Open Catalogue , a marketplace where the users and user communities can browse, share, store and download ready to use machine learning and deep learn-
ing modules. This includes working and ready to use applications (such as image classification tools, network anomaly detection engines, etc.) as well as more general purpose and generic models.

The DEEP learning facility that coordinates and orchestrates the overall model training, testing and evaluation, choosing the appropriate Cloud resources according to the computing and storage

resources.

The DEEP as a Service solution (DEEPaaS), that provides a way to deploy and serve an already trained model, stored in the catalogue, as a service, to provide third users with the model functionality and acquired knowledge.

The storage and data services where the user data, training and validation results, as well as any other data assets are stored.Many cloud architectures have been developed depending upon the ML requirement.

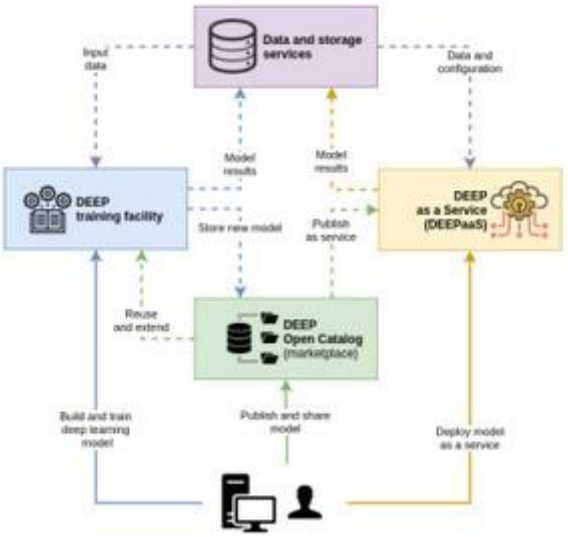

Fig 5

Cloud architecture for analyzing real-time road traffic data:

With the expanded usage of IoT sensors in road traffic and an increasing number of vehicles on the road, analyzing road traffic data is something to be considered. Road traffic data is collected using IoT devices and these datasets are massive.The current IoT systems are based on simple sensors that collect data to a cloud and actuators controlled by applications in the cloud . The cloud provides access to virtually unlimited resources that can be programmatically provisioned with a pay-as-you-go pricing model, enabling applications to elastically adjust their deployment topology

Road traffic data is collected using IoT sensors. There are many different IoT sensor types that are used in traffic monitoring, for example light detectors, radars, cameras, humidity sensor, accelerometer, fire sensor, etc. Those sensors collect road traffic data, which are gathered in the

IoT gateway. The IoT gateway is used as a bridge between IoT sensors and the cloud. It is used for initial data filtering.Kafka is an open-source distributed event streaming platform. It is used for high-performance data pipelines, streaming analytics, data integration, and mission-critical applications. It is scalable, has high availability and has high throughput . All sensor data is stored in the Kafka cluster. As an intermediary between IoT sensors and Kafka MQTT servers are used, accessible through MQTT load balancer. This will allow easy horizontal scaling of MQTT Servers. MQTT is a standard messaging protocol for IoT.This Methodology provides a robust architecture for analyzing streaming data

A cloud-to-edge architecture for predictive analytics:

Data management and processing to enable predictive analytics in cyber physical systems, holds the promise of creating insight into the underlying processes, discovering criticalities and predicting imminent problems. Hence, proactive strategies can be adopted, with respect to predictive analytics. This paper discusses the design and prototype implementation of a plug-n-play end-to-end cloud architecture, enabling predictive maintenance of industrial equipment. This is enabled by integrating edge gateways, data stores at both the edge and the cloud, and various applications, such as predictive analytics, visualization and scheduling, integrated as services in the cloud system. The proposed approach has been implemented into a prototype and tested in an industrial use case related to the maintenance of a robotic arm.The advent of Industry 4.0 trend in automation and data exchange, leads to a constant evolution towards smart environments, including an intensive utilization of Cyber-Physical System (CPS).This promotes a full integration of manufacturing IT and control systems with physical objects embedded with electronics, software and sensors. This new industrial model leads to a pervasive integration of information and communication technology into productive components, generating massive amounts of data. Powerful and reliable cyber-physical architectures are becoming prominent to effectively analyze such large amounts of data, creating insight into the production process, and, thus, enabling its improvement, as well as competitive business advantages.

The SERENA system comprises a number of services, which collectively provide predictive analytics functionality, enabling predictive maintenance policies to be applied. It is implemented

using a light-weight micro-services architecture, which utilizes Docker containers to wrap the services into deployable units. The services can then be distributed across the SERENA hybrid cloud, extending their functionality out to edge gateways on the factory floor. The distribution of services and dynamic communications channels is implemented using a Docker orchestration manager. Wrapping services in containers abstracts them from the underlying host infrastructure. As Docker is a commonly supported open source technology, the SERENA system can be deployed on a wide variety of infrastructures, from hardware servers and gateways, through virtual machines, to hosted environments on public and hybrid clouds. Using the same Docker solution across the SERENA hybrid cloud and gateways, gives the system a unified architecture, which can be operated and managed as a single unit. The services represent logical elements that provide defined functionality in the SERENA system. Whilst the SERENA reference implementation uses specific technology to realize each service, the common interface allows technology to be swapped, depending on the specific implementation requirements. This technology transparency is an important concept in SERENA's plug-n-play architecture. SERENA architecture is shown in fig 6.This architecture can be regarded a first step in Industry 4.0

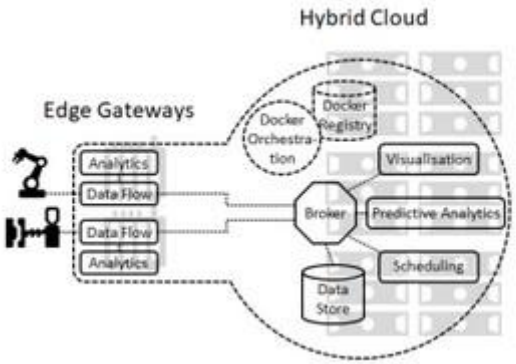

Fig 6

Architecture of Sensor-Cloud:

Cloud computing service framework provides shared network services to customers who are not concerned with the implementation details of the services given to them. The service instances (e.g., virtual sensors) generated by cloud computing services are automatically provisioned to users when they make a request.

Routing, clock synchronisation, data processing, power management, OS, location, and programming were all topics covered in prior studies on physical sensors. Few studies, however, focus on physical sensor management since these physical sensors are inextricably linked to their specialised application as well as their direct users. Users, however, can't use these physical sensors directly when they need them, unless they're using their respective sensor services.

Therefore, these physical sensors should be supervised by some special sensor-management schemes. The Sensor-Cloud infrastructure would subsidize the sensor system management, which ensures that the data-management usability of sensor resources would be fairly improved.

A hybrid edge-cloud architecture for reducing on-demand gaming latency:

Cloud gaming, also known as gaming on demand or gaming-as-a-service, is a form of internet gaming in which video games are hosted on remote servers and streamed directly to a user's device, or, more informally, playing a game from the cloud. It differs from traditional gaming, in which a user's video game console, personal computer, or mobile device executes the game locally. It is demonstrated through a large-scale measurement study that the existing cloud infrastructure is unable to meet the strict latency requirements necessary for acceptable on-demand game play. It is found that a hybrid infrastructure significantly improves the number of end-users served.

On-demand gaming faces two major technical challenges: latency and the necessity for servers with expensive, specialised technology that can't service numerous gaming sessions at the same time. On-demand gaming suffers from encoding latency, which is the time it takes to compress the video output, and network latency, which is the time it takes to transport user input and video output back and forth between the end-user and the cloud, due to offloading computing to a remote host. Although better hardware encoders will certainly reduce video encoding latency,

network latency is unavoidable because it is limited by the speed of light in fibre. According to previous research, gamers begin to notice a delay of 100 milliseconds.

IaaS Cloud Architecture: From Virtualized Datacentres to Federated Cloud Infrastructures: Datacentres have evolved from expensive, rigid, mainframe-based architectures to agile distributed architectures based on commodity hardware that developers can dynamically shape, partition, and adapt to different business processes and variable service workloads. Future datacentres will look like private IaaS clouds, supporting the flexible and agile execution of virtualized services. In this context, the virtual infrastructure manager, also called the cloud operating system (cloud OS), orchestrates the deployment of virtual resources and manages the physical and virtual infrastructures to command-and-control service provisioning. This cloud vision of the datacentre represents not only a new provisioning model but also a way to simplify and optimize infrastructure operation because applications are not tied to a specific physical server and data is not attached to a single storage device. This provides several advantages from the infrastructure management perspective, including server consolidation to reduce hardware and power requirements.

There are many similarities between a typical computer system's threaded OS and a cloud OS. A computer OS's main role is to manage the computer resources—the CPU, memory, disks, and I/O devices—and provide a secure and isolated multithreaded execution environment for user applications. Similarly, a cloud OS's role is to efficiently manage datacentre resources to deliver a flexible, secure, and isolated multitenant execution environment for user services that abstracts the underlying physical infrastructure and offers different interfaces and APIs for interacting with the cloud. To provide an abstraction of the underlying infrastructure technology, the cloud OS can use adapters or drivers to interact with a variety of virtualization technologies. These include hypervisor, network, storage, and information drivers. The core cloud OS components, including the virtual machine (VM) manager, network manager, storage manager, and information manager, rely on these infrastructure drivers to deploy, manage, and monitor the virtualized infrastructures.

Cloud federation, which enables cloud providers and IT companies to collaborate and share their resources, is associated with many portability and interoperability issues. Cloud developers and researchers have proposed or implemented numerous federation architectures, including

cloud bursting, brokering, and aggregation. These architectures can be classified according to the level of coupling:

Loosely coupled federation: This scenario is formed by independent cloud instances—for example, a private cloud complementing its infrastructure with resources from an external commercial cloud—with limited interoperation between them.

Partially coupled federation: This scenario typically consists of various partner clouds that establish a contract or framework agreement stating the terms and conditions under which one partner cloud can use resources from another.

Tightly coupled federation: This scenario usually includes clouds belonging to the same organization and is normally governed by the same cloud OS type.

The definition of an architecture reference model for IaaS clouds is essential for the widespread adoption of these technologies. The core component of this architecture—the cloud OS—is responsible for managing and monitoring the physical and virtual infrastructures, providing abstraction of the underlying infrastructure, and offering different tools and advanced functionality for cloud users. In addition, the cloud OS must offer federation capabilities to allow IT companies to scale out their local datacentres with external resources or to share and aggregate resources with partner infrastructures to increase computing capacity and reduce costs.

Prognostics As-A-Service: A Scalable Cloud Architecture for Prognostics:

Comprehensive aircraft system health-state awareness is critical for maintaining safe, efficient growth in global operations as well as enabling higher levels of autonomy and new forms of aviation. Maintainers, operators, controllers, dispatchers, pilots, autonomous systems, and other decision makers must have reliable real-time knowledge of the vehicle health, health of its critical composite systems, predictions of how health changes with time, and predictions of how its capabilities change with health degradation in order to preserve safety and efficiency. Providing this information reliably in computationally constrained environments and across the wide range of vehicles and systems continues to be a challenge.

NASA is developing Prognostics As-A-Service, a cloud computing service for diagnostics and prognostics. The Prognostics As-A-Service effort at NASA explores the feasibility and challenges of cloud-enhanced prognostics. Though such a system has wide applicability, this research effort was focused on aviation applications. This effort is exploring and demonstrating

the ability to address the major challenges of a PaaS architecture. This paper details the PaaS architecture and describes its use in NASA projects.

PaaS is designed to fulfil a need for in-flight prognostics beyond what can be achieved onboard. The PaaS service augments any onboard prognostics by applying more powerful hardware to enable more sophisticated prognostic algorithms. The service covered by this paper includes facilities to register and manage platforms, systems and components, configure those entities, send data to the service, and retrieve events from the service. As part of the service's operation, it naturally stores most data received and calculated in a database.

The Prognostics as a Service architecture builds on the Generic Software Architecture for Prognostics (GSAP) library. PaaS provides the infrastructure to store and manage data and configuration information associated with systems being prognosed, and to efficiently pass data to prognosers and results back to the user. The prototype system consists of a Prognostics Application Driver written in C++ that wraps GSAP in a thin communication framework that talks to the front-end process. The front-end process consists of a web server written in Java that exposes a RESTful API to end-users. The application also uses a PostgreSQL database to store prognostics data, configuration information, and application state. It also monitors the message bus for prognostic events and passes those events back to the Java server application.

The Service Layer, contains components that execute the main "business logic" of the system. This includes input validation, storage and retrieval of data from the database. The service layer also routes session data needed by the prognostics component to the prognostics component in the execution layer.

The REST API Layer, is a web architecture that allows for both querying and updating of resources using structured HTTP requests. The REST style provides a uniform stateless architecture that fits well into today's web-centric world. The API is not without limitation however. In particular, the REST format (and HTTP in general) did not provide any mechanism for push-based notifications.

Due to this limitation, the current API requires the clients poll periodically to receive new events. Feedback from the CASAS and SWS projects show a potential for improvement in usability when addressing domains outside of component health management. Using more generalized system models has its advantages in being applicable to many domains, but also increases the burden of translating inputs and outputs to something useful. Perhaps allowing an end user to map a
domain-specific ontology onto the generic PaaS interface would allow for a simplification of

integration and promote further use, possibly in novel ways. Further feedback from the CASAS project suggests that connecting vehicles directly to internet services is likely to receive pushback from those concerned with aircraft safety. The SWS project will continue to mature their safety metrics and safety assessment algorithms in the PaaS platform. The PaaS architecture will be matured to better support these advancements.

CHAPTER 3

3.1) Methodologies/Techniques/Algorithms:

Workload placement algorithm - Workload placement is a term that has developed in reference to managing resources in hardware virtualization systems. It involves the assignment of workloads in such a way that VM efficiency is optimized. Workload placement involves assigning a specific workload where it can be easily completed by a VM or another component. Some issues involved here include the allocation of resources, such as how a virtual CPU or a virtual memory is assigned to a VM.

When there is only one server at each tier of the edge cloud, the workload placement problem is formulated as a mixed nonlinear integer programming (MNIP) problem, with integer variables indicating the locations of workloads being placed and non-integer variables indicating the amount of capacity allocated to execute each workload. Then, we solve this MNIP problem in two steps. First, we fixed the integer variables and transformed the MNIP problem to a convex optimization problem. Second, based on the non-integer variable being obtained in the first step, we further solve a combinatorial optimization problem which only contains the integer variables.

Algorithm 1: The simulated annealing process

Randomly generate an initial solution $\mathbf{x}_{old}$.
$T \leftarrow T_{max}$
while $T > T_{min}$ **do**
    Generate a new solution $\mathbf{x}_{new}$
    $\Delta d = f(\mathbf{x}_{new}) - f(\mathbf{x}_{old})$
    if $\Delta d < 0$ **then**
        $\mathbf{x}_{old} \leftarrow \mathbf{x}_{new}$
    end
    else if $Rand(0,1) < exp(-\frac{\Delta d}{T})$ **then**
        $\mathbf{x}_{old} \leftarrow \mathbf{x}_{new}$
    end
    $T \leftarrow \alpha \cdot T$
end

Table 1: Technical comparison of different messaging approaches and algorithms being used in several papers of Sensor-Cloud.

Papers	Using traditional SOAP messages	Using SPWS messages	Using data caching or hop-by-hop processing	Using QoS-based routing	Using SGIM	Using SPMC	Using event-matching algorithms	Using matrix-matching algorithms	Using orthogonal neural network algorithm	Using load-balancing algorithm
[59]	No	No	No	No	Yes	Yes	Yes	No	No	No
[5]	No	No	Yes	Yes	No	Yes	Yes	No	No	No
[34]	No	Yes	No	No	No	No	No	No	No	No
[60]	No	No	Yes	No	No	No	Yes	No	No	No
[61]	No	No	No	Yes	No	No	Yes	No	No	No
[39, 62]	No	No	No	No	No	No	Yes	Yes	No	No
[13, 63]	No	No	Yes	No	No	No	No	No	No	No
[64]	No	No	No	No	No	No	No	No	Yes	No
[65]	No	No	Yes	Yes	No	No	No	No	No	No
[37, 66, 67]	Yes	No	No	No	No	No	No	No	No	No
[45]	Yes	No	No	No	No	No	No	No	No	Yes

SPWS: sensor profile for web services.
SGIM: statistical group index matching.
SPMC: stream monitoring and processing component.

CHAPTER 4

4.1) Analysis

MTCEM is a security duty separation model for cloud computing services, with the goal of implementing a fair co-management method between CSP and cloud consumers on CSP-owned cloud computing platforms. MTCEM's key contribution is that it introduces a novel two-level hierarchy transitive trust chain method to suit cloud users' demand for control over cloud computing platforms.

Cloud computing security problems should be understood by organisations who are implementing cloud computing by expanding their on-premise infrastructure. A defence in depth technique must be used to protect against the compromising of compliance integrity and security of their apps and data. Firewall, intrusion detection and prevention, integrity monitoring, log inspection, and virus protection are all part of this line of defence. Proactive enterprises and service providers should implement this protection on their cloud infrastructure to accomplish security and get an advantage over their competitors by utilising cloud computing. A physical cloud computing security architecture is proposed in this paper.

Formal analysis and careful experimentation highlight the advantages of this hierarchical architecture over typical flat edge cloud. To efficiently execute mobile programmes in the edge cloud, we continue to develop workload placement algorithms. Trace-based simulations are used to verify the effectiveness of the proposed technique.

As a result, Cloud applications can be extended to the smart home domain, allowing more users to profit from Cloud and the Cloud provider to reach a larger audience. This architecture uses gateway technology to create a link between the Cloud and the smart home, allowing the smart home to be merged into the Cloud to give more information and access to Cloud-based services.

Cloud gaming or the gaming-as-a-service is often referred to as online gaming that has the ability to run video games on remote servers and stream the ongoing action directly to a user’s screen. The user need not possess any kind of graphical unit to run any intensive forms of games instead the user can just adjust with a stable and secure internet connection. Cloud gaming allows users to run the kind of game they desire in an untraditional way wherein a game can run locally without the user’s game console, personal computer, or mobile device.

The hierarchical architecture of edge cloud enables aggregation of the peak loads across different tiers of cloud servers to maximize the amount of mobile workloads being served. To ensure efficient utilization of cloud resources, it is proposed to have a workload placement algorithm that decides which edge cloud servers mobile programs are placed on and how much computational capacity is provisioned to execute each program. The performance of the proposed hierarchical edge cloud architecture on serving mobile workloads is evaluated by formal analysis, small-scale system experimentation, and large scale trace-based simulations.

A computational model of a multilayer architecture for increasing the performance of devices

using the Mobile Cloud Computing paradigm was built. The main novelty of the work lies in defining a comprehensive model where all the available computing platforms along the network layers are involved to perform the outsourcing of the application workload. This proposal provides a generalization of the Mobile Cloud Computing paradigm which allows handling the complexity of scheduling tasks in such complex scenarios.

The Cloud applications are extended to the domain of smart home so that more users can benefit from Cloud and the Cloud provider can have more potential users. By utilizing gateway technology, this architecture builds a bridge between Cloud and smart home so that smart home can be merged into Cloud to provide more information services and access services provided by Cloud.

CONCLUSION

Cloud computing offers us with infinite computing capabilities, scalability, pay-per use scheme, etc. From these papers, we understand how cloud computing is the development trend of the future. Since, our data is public when hosted on the cloud, it is vulnerable to malicious attacks.Organizations that are implementing cloud computing by expanding their on-premise infrastructure, should be aware of the security challenges faced by cloud computing. Some of these papers have introduced us to much more sophisticated and secure cloud computing architectures that can be used in the near future for better security and user experience. Certain models like, MTCEM presents a new mechanism of two-level hierarchy transitive trust chain to meet the requirement of cloud customers to control the computing platforms in the cloud.

We also understand how the performance of mobile computing could be significantly improved by leveraging cloud computing and migrating mobile workloads for remote execution at the cloud. An hierarchical edge cloud architecture can improve the efficiency of cloud resource utilization in this aspect. We also come across a computational model of a multilayer architecture for increasing the performance of devices and handling the complexity of scheduling tasks using the Mobile Cloud Computing paradigm.

We also come across cloud architectures based on smart home, so that a smart home can be merged into cloud to provide more information services and access services from the cloud. Similarly, IoT sensors which are used for road traffic management do load the cloud with immense data. A cloud architecture is proposed which is scalable, reliable and available. This architecture can be used with various sensors and for different analysis. It can return various recommendation to the road traffic users.

Distributed architectures to provide machine learning practitioners with a set of tools and cloud services that cover the whole machine learning development cycle are also proposed. DEEP-Hybrid-DataCloud framework allows transparent access to existing e-Infrastructures, effectively exploiting distributed resources for the most compute-intensive tasks coming from the machine learning development cycle.

These papers have educated us with several cloud architectures that can be used in different domains. This have given us a better understanding of the current architectures that are being implemented. We realized the flaws of the current cloud technologies. To conclude, Cloud computing is the latest technology that promises immense benefits however there is lot of research which is still required in this area as many of the concerns related to security and privacy issues are not been answered by the experts and remains open for research and development.By considering the contributions from several IT industries worldwide, it’s obvious that cloud computing will be one of the leading strategic and innovative technologies in the near future.

REFERENCES

1. Cloud Computing: Concepts, Architecture and Challenges
2. Cloud Computing: Architecture and Challenges
3. A trusted computing environment model in cloud architecture
4. Secure cloud architecture
5. A hierarchical edge cloud architecture for mobile computing
6. Multilayer architecture model for mobile cloud computing paradigm
7. A cloud architecture based on smart home
8. Cloud architecture for analyzing real-time road traffic data
9. A cloud-based framework for machine learning workloads and applications
10. A cloud-to-edge architecture for predictive analysis
11. Sensor-cloud: architecture, application and approaches
12. SASC: secure and authentication-based sensor cloud architecture for intelligent internet of things
13. A hybrid edge-cloud architecture for reducing on-demand gaming latency
14. IAAS cloud architecture: from virtualized datacenters to federated cloud infrastructures
15. Prognostics as-a-service: a scalable cloud architecture for prognostics