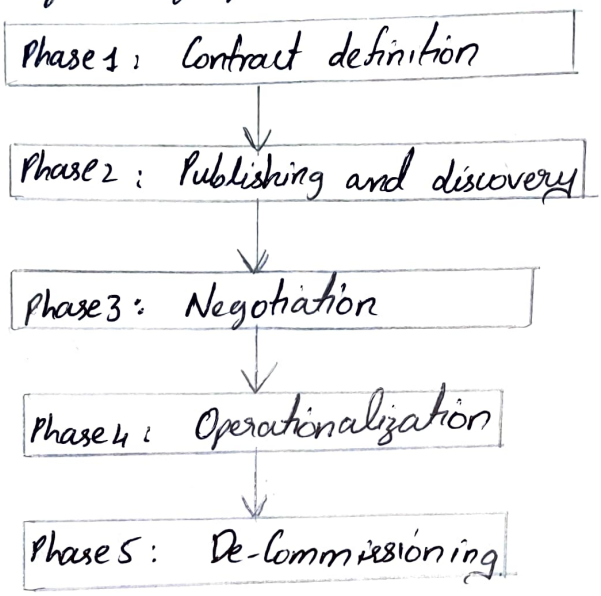
CC ASSIGNMENT-2

R.SAI KIRAN
18N31A05J4
CSE-D

UNIT-V

① **LIFE CYCLE OF SLA:**

- Each SLA goes through a sequence of steps starting from identification of terms and conditions.
- Activation and monitoring of the stated terms and conditions.
- Eventual termination of contract once the hosting relationship ceases to exist.
- Such a sequence of steps is called SLA Life cycle.



(i) Contract Definition:

Generally service providers, define a set of service offerings and corresponding SLA's using standard templates.

These service offerings form a catalog.

Individual SLA's for enterprises can be derived by customizing these SLA templates.

(ii) Publication and Discovery:

Service provider advertises these base service offerings through standard publication media.

The customers should be able to locate the service provider

by searching the catalog.

The customers can search different competitive offerings and shortlist a few that fulfill their requirements for further negotiation.

(iii) Negotiation: Once the customer has discovered a service provider who can meet their application hosting record need. The SLA terms and conditions needs to be mutually agreed upon before signing the agreement for hosting the application.

for a standard packaged application which is offered as service, this phase could be automated.

(iv) Operationalization:

SLA operation consists of SLA monitoring, SLA accounting and SLA enforcement.

SLA monitoring: It involves measuring parameter values and calculating the metrics defined as a part of SLA and determining the deviations.

SLA accounting: The application's actual performance and the performance guaranteed as a part of SLA is reported.

SLA enforcement: It involves taking appropriate action when the runtime monitoring detects a SLA violation.

(v) De-Commissioning:

SLA decommissioning involves termination of all activities performed under a particular SLA.

When the hosting relationship between the service provider and the service consumer has ended,

2 TYPES OF SLA:

- Service level agreement provides a framework within which both seller and buyer of a service can pursue a profitable service business relationship.
- It outlines the broad understanding between the service provider and the service consumer for maintaining a mutually beneficial relationship.
- SLA can be modeled using web service-level agreement language specification.

→ Key components of a SLA

Service level Parameter	Describes an observable property of a service whose value is measurable.
Metrics.	Metrics are the key instrument to describe exactly what SLA parameters mean by specifying how to measure or compute the parameter values.
Function.	Functions are central to describing exactly how SLA parameters are computed from resource metrics.
Measurement directives	These specify how to measure a metric.

→ Two types of SLA's from the perspective of application hosting.

(1) Infrastructure SLA:

- The infrastructure provider manages and offers guarantees on availability of the infrastructure, namely, server machine, power, network connectivity.

- Enterprise mange themselves, their applications that are deployed on these server machines.

(ii) Application SLA:

- In the application co-location hosting model, the server capacity is available to the applications based solely on their resource demands.
- The service providers are flexible in allocating and de-allocating computing resources among the co-located applications.
- Therefore, the service providers are also responsible for ensuring to meet their customer's application SLO's.
- The SLA perspective there are multiple challenges for provisioning the infrastructure on demand.

→ These challenges are as follows.

(a) The application is a black box to the managed service provider (MSP) and the MSP has virtually no knowledge about the application runtime characteristics.

(b) The MSP needs to understand the performance bottlenecks and the scalability of the application.

(c) The MSP analyzes the application before it goes on-live.

(d) If every customer decides to select the highest grade of SLA simultaneously, there may not be a sufficient number of servers for provisioning.

13 SLA MANAGEMENT IN CLOUD :-

SLA management of applications hosted on cloud platforms involves five phases.

- Feasibility.
- On-boarding
- Pre-production
- Production
- Termination.

(i) Feasibility Analysis :

- MSP conducts the feasibility study of hosting an application on their cloud platforms.
- This study involves three kinds of feasibility.

- Technical feasibility:
  - Ability of an application to scale out.
  - Compatibility of the application with the cloud platform being used within the MSP's data center.
  - The need and availability of specific hardware and software required for hosting and running of the application.
  - Preliminary information about the application performance and whether they can be met by the MSP.

- Infrastructure feasibility:

It involves determining the availability of infrastructural resources in sufficient quantity so that the projected demands of the application can be met.

- Financial feasibility:

The study involves determining the approximate cost to be incurred by the MSP and the price the MSP charges the

customer so that the hosting activity is profitable to both.

A feasibility report consists of the results of the above three feasibility studies.

(ii) On-Boarding of Application:

Once the customer and the MSP agree in principle to host the application based on the findings of feasibility study, the application is moved from the customer servers to the hosting platform.

On-boarding activity consists of following steps.

(a) Packing of the application for deploying on physical or virtual environments.

(b) The packaged application is executed directly on the physical servers to capture and analyze the application performance characteristics.

(c) The application is executed on a virtualized platform and the application performance characteristics are noted again.

(d) Based on the measured ~~plat~~ performance characteristics, different possible SLA's are identified.

(e) Once the customer agrees to the set of SLO's and the cost, the MSP starts creating different policies required by the data center for automated management of the application.

(iii) Pre-Production:

Once the determination of policies is completed as discussed in previous phase, the application is hosted in a simulated production environment.

It facilitates the customer to verify and validate the MSP's findings on application's runtime characteristics and agree on

the defined SLA.

(iv) Production:

- In this phase, the application is made accessible to its end users under the agreed SLA.
- However, there would be situations when the managed application tends to behave differently in a production environment compared to the preproduction environment.

(v) Termination:

- When the customer wishes to withdraw the hosted application and does not wish to continue to avail the services of the MSP for managing the hosting of its application, the termination activity is initiated.
- On initiation of termination, all data related to the application are transferred to the customer and only the essential information is retained for legal compliance.

4.

4 TRADITIONAL APPROACHES TO SLO MANAGEMENT:

The SLO's can be referred to as measurable characteristics of an SLA, such as Quality of Service (QoS) aspects that are achievable, measurable, meaningful and acceptable for both service providers and customers.

Traditionally, load balancing techniques and admission control mechanisms, have been used to provide guaranteed quality of service (QoS) for hosted Web applications.

⇒ LOAD BALANCING:

The objective of a load balancing is to distribute the incoming requests onto a set of physical machines.

The load balancing algorithm executes on a physical machine that interfaces with the clients.

- Class-agnostic: The front-end node is neither aware of the type of client from which the request originates nor aware of the category to which the request belong to.
- class-aware: The front-end node must additionally inspect the type of client making the request and is the type of service requested before deciding which back-end should service the request.

=> Admission Control:

Admission control play an important role in deciding the set of requests that should be admitted into the application server, when the server experiences "very" heavy loads.

During overload situations, since the response time for all the requests that should be admitted into the server.

It would be preferable to be selective in identifying a subset of requests that should be admitted into the system so that the overall pay-off is high.

- Request-based mechanism:

It reject new requests if the servers are running to their capacity.

The disadvantage with this approach is that a client's session may consist of multiple requests that are not necessarily unrelated.

- Session-based mechanism:

It try to ensure that longer sessions are completed and any new sessions are rejected.

Accordingly, once a session is admitted into the server, all future requests belonging to that session are admitted as well, even though new sessions are rejected by the system.

5 SLA MANAGEMENT IN CLOUD COMPUTING:

- A service level agreement (SLA) is a commitment between provider and a client.
- Particular aspects of the service, such as quality, availability, responsibilities are agreed upon between the service provider and the service user.
- In the early days of web-application deployment, performance of the application at peak load was a single important criteria for provisioning server resources.
- The web applications were hosted on these dedicated individual servers within enterprises own server rooms.
- These web applications were used to provide different kinds of e-services to various clients.
- The service-level objectives (SLO's) for these applications were response time and throughput of the application end-user requests.
- The activity of determining the number of servers and their capacity that could satisfactorily serve the application end-user requests at peak loads is called capacity planning.
- The planned capacity for each of the applications to run successfully in three servers.
- As the number of web applications grow, the server rooms in the organization became large and such server rooms were known as data centers.
- These data centers were owned and managed by the enterprise themselves.

- The enterprises need not invest in procuring expensive hardware upfront without knowing the viability of the business.
- As the number of web applications grow, the level of sophisti-cation required to manage the data centers increased.
- The QoS parameters are related to the availability of the system CPU, data storage, and network for efficient execution of the application at peak loads.

UNIT-IV

1 SOFTWARE AS A SERVICE (SaaS):

- SaaS is a model of software deployment where an application is hosted as a service provided to customers across internet.
- SaaS reduce the burden of software maintenance support, but users have responsibility control over software versions and requirements.
- SaaS is at highest layer and features a complete application offered as a service.

SaaS Maturity Model:

Level 1: Ad-Hoc / Custom One instance per customer.

Level 2: Configurable per customer.

Level 3: Configurable & Multi Tenant Efficient.

Level 4: Scalable, Configurable & Multi Tenant Efficient.

⇒ Single Tenant:

- In a single tenant model each of the tenants will get their own respective instances.
- There is a absolutely no sharing of anything (code, DB, etc.).
- Each time a new customer is added a new (logical) hardware is provisioned and new instance of the product is setup in the allocated environment.
- Maturity level 1 and 2 falls under single tenant model.

advantages:

- Takes less time to roll-out.
- Overall SaaS transition complexity and cost is going to be less.
- Does not require any SaaS expertise.
- It supports non-web native applications.

disadvantages:

- Maintenance efforts are going to be huge as you will have to maintain multiple code environments.

⇒ Multi-Tenant:

- Multi Tenancy is an architecture capability that allows an application to recognize tenants and exhibit functionalities as per the configuration set for the tenants.
- An on-premise application is typically designed to work for a single organization.
- Maturity level 3 and 4 fall under the multi tenant model.

advantage:

- facilitates a cost effective way of delivering SaaS solution.
- Huge savings in operational cost, particularly over long run.
- Roll out of upgrades/fixes is much easier.

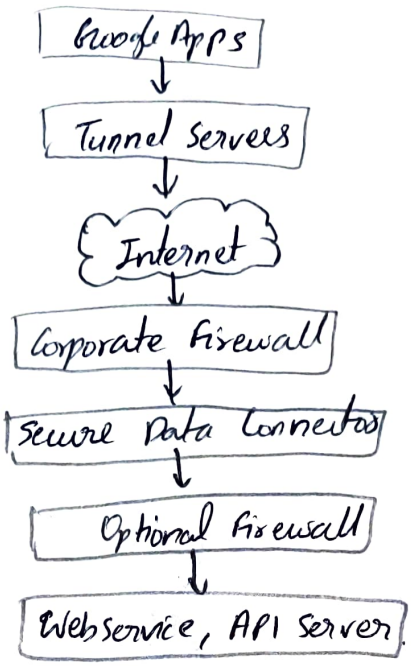
disadvantage:

- Initial investment to enable multi tenancy is high.
- Requires SaaS architecture expertise, which may not be within the company.

---

2 GOOGLE APP ENGINE:

- The google app engine is a cloud-based platform.
- It is quite comprehensive and combines IaaS, PaaS and SaaS.
- The app engine supports the delivery, testing and development of software on demand in a cloud computing environment
- The company extends its platform and infrastructure to the cloud through its app engine.
- It is a platform for hosting web applications in google ~~man~~ managed data centers.
- It is cloud computing technology which virtualizes applications across multiple servers and data centers.
- It presents the platform to those who want to develop SaaS solutions at competitive costs.



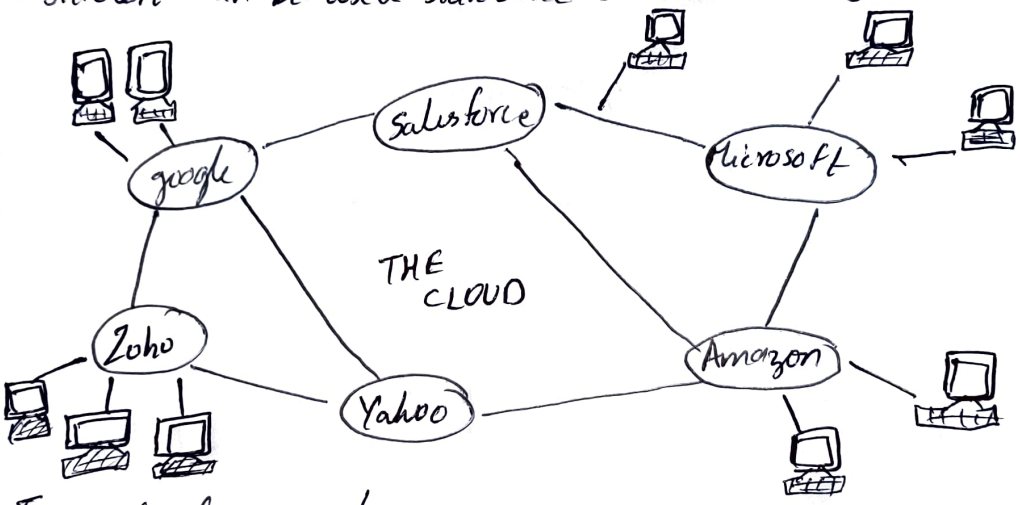
- The SDC constructs an encrypted connection between data source and google appls.
- When the user wants to get the data, he/she will first send an authorized data requests to google apps, which forwards the request to the tunnel server.
- The tunnel servers validate the request identity.
- If the identity is valid, the tunnel protocol allows the SDC, to set up a connection, authenticate and encrypt the data flows across the internet.
- At the same time, the SDC uses resources rules to validate whether a user is authorized to access a specified resource.
- When the request is valid, the SDC performs a network request.
- The uploading session can only ensure that the data received by the cloud storage is the data that the user uploaded.
- The downloading session can guarantee the data that the user retrieved is the data cloud storage recorded.

3 SaaS Integration :

- Cloud-centric integration solutions are being developed and demonstrated for showcasing their capabilities for integrating enterprise and cloud applications.
- Composition and collaboration will become critical and crucial for the mass adoption of clouds.

Jitterbit :

- Jitterbit is a fully graphical integration solution that provides users a versatile platform suite of productivity tools to reduce the integration efforts sharply.
- Jitterbit can be used standlone or with existing EAI infrastructure.



Two major Components :

(i) Jitterbit Integration Environment :

An intuitive point-and-click graphical UI that enables to quickly configure, test, deploy and manage integration projects on the jitterbit server.

(ii) Jitterbit Integration Server :

A powerful and scalable run-time engine that processes all the integration operations, fully configurable and manageable from the Jitterbit application.

4.

[4] CLOUD COMPUTING and IDENTITY:

Digital identity:

- It holds the key to flexible data security within a cloud environment.
- A digital identity represents who we are and how we interact with others on-line.
- Access, identity and risk are three variables that can become inherently connected when applied to the security of data, because access and risk are directly proportional: as access increases, so then risk to the security of data increases.
- access controlled by identifying the actor attempting the access is the most logical manner of performing this operation.
- Ultimately, digital identity holds the key to securing data, if that identity can be programming linked to security policies controlling the post-access using of data.

Identity, Reputation and Trust:

- Reputation is a real-world commodity; that is a basic requirement of human-to-human relationships.
- Our basic societal communication structure is built upon the idea of reputation and trust.
- Reputation and its counter value, trust is bec easily transferable to a digital real moeBay, for example, having party built a successful business model on the strength of ratings system builds up the reputation of its buyers and sellers through successful transactions.
- These types of reputation systems can be extremely useful when used with a digital identity.
- They can be used to associate varying levels of trust with that identity, which in turn can be used to define the level.

→ Using Information Cards to Protect Data:

- Information cards are built around a set of open standards devised by a consortium that includes Microsoft, IBM, Novell and soon.
- The original remit of the cards was to create a type of single sign on system for the internet, to help users to move away from the need to remember multiple passwords.
- However this information cards system can be used in many more ways.
- Because an information card is a type of digital identity, it can be used in the same way that other digital identities can be used.
- For example, an information card can be used to digitally sign data and content and to control access to data & content.

5 CLOUD COMPUTING and DATA SECURITY RISK:

- Cloud computing and data security risk is a development that is meant to allow more open accessibility and easier and improved data sharing.
- Data are uploaded into a cloud and stored in a data center, for access by users from that data center; or in a more fully cloud-based model, the data themselves are created in the cloud and stored and accessed.
- The most obvious risk in this scenario is that associated with the storage of that data. A user uploading or creating cloud-based data include those data that are stored and maintained by a third party cloud povider such as google, amazon, Microsoft and so on.

→ This action has several risks associated with it:

- Firstly, it is necessary to protect the data during upload into

data center to ensure that the data do not get hijacked on the way into the database.

- Secondly, it is necessary to the stores the data in the data center to ensure that they are encrypted at all times.
- Thirdly, and perhaps less obvious, the access to those data need to be controlled; this control should also be applied to the hosting company, including the administrators of the data center.

- Data security risks are compounded by the open nature of cloud computing.
- Access control becomes a much more fundamental issue in cloud-based systems because of the accessibility of the data.
- A further area of risk associated not only with cloud computing but also with traditional network computing, is the use of content after access.
- The risk is potentially higher in a cloud network, for the simple reason that the information is outside of your corporate walls.

——— *THE END* ———