SMSCLOUD: A HYBRID ARCHITECTURE USING MULTIPLE CLOUDS

S.Jabeen Begum
Research scholar
Velalar College of Engg. & Tech.
Erode – 12
Email: sjabeenbegum@yahoo.co.in

Dr.T.Purusothaman
Asst.Professor/CSE
Government College Tech.
Coimbatore – 13
Email:purushgct@yahoo.com

Ebenezer Princy.R, Vidhya.G
II[nd] Year M.E CSE
Velalar College of Engg & Tech.
Erode – 12
Email: princy_kits@yahoo.co.in
Email: gvidhyacse@gmail.com

ABSTRACT

Cloud computing is a topic of intense interest in the Internet field. Major IT giants have launched their own cloud computing products. Cloud computing focuses on delivery of reliable, secure, fault-tolerant, sustainable, and scalable infrastructures for hosting Internet-based application services. This unique paradigm brings about the research challenges, which have not been well explored. In this article named SMSCloud(Secured, managed and Scalable Cloud), we present our solutions to security without affecting performance, admission control, scalability, SLA negotiation. Here we focus on demand forecasting scheme and effective billing mechanism.

Keywords:- Cloud computing, Demand forecasting, admission control.

1. INTRODUCTION

Cloud Computing, the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making software even more attractive as a service and shaping the way IT hardware is designed and purchased. Developers with innovative ideas for new Internet services no longer require the large capital outlays in hardware to deploy their service or the human expense to operate it. They need not be concerned about over provisioning for a service whose popularity does not meet their predictions, thus wasting costly resources, or under provisioning for one that becomes wildly popular, thus missing potential customers and revenue. Moreover, companies with large batch-oriented tasks can get results as quickly as their programs can scale, since using 1000 servers for one hour costs no more than using one server for 1000 hours. This elasticity of resources, without paying a premium for large scale, is unprecedented in the history of IT.

Cloud Computing Uses:

• Five characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service.

• Four deployment models: private clouds, community clouds, public clouds, and hybrid clouds.

• Three service models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).

Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS). The datacenter hardware and software is what we will call a Cloud. When a Cloud is made available in a pay-as-you-go manner to the general public, we call it a Public Cloud; the service being sold is Utility Computing. We use the term Private Cloud to refer to internal datacenters of a business or other organization, not made available to the general public. Thus, Cloud Computing is the sum of SaaS and Utility Computing, but does not include Private Clouds. People can be users or providers of SaaS, or users or providers of Utility Computing.

Here we focus on the issues of both cloud users and cloud providers in the following aspects:

1. The user may be reliable or unreliable, so the provider has to focus on which user is trustworthy.
2. What kind of security can be given for cloud users for his data?
3. When the demand exceeds the resource level how will the cloud providers manage?
4. Provide a billing system such that the user meets his exact demand.

2. RELATED WORK

Grid computing must be acknowledged as an intellectual sibling of, if not ancestor to, cloud computing [1, 2, 3, 4]. The original metaphor for a computational utility, in fact, gives grid computing its name. While grid computing and cloud computing share a services oriented approach [5, 6] and may appeal to some of the same users (e.g., researchers and analysts performing loosely-coupled parallel computations), they differ in two

key ways. First, grid systems are architected so that individual user requests can (and should) consume large fractions of the total resource pool [7]. Cloud systems often limit the size of an individual request to be tiny fraction of the total available capacity [8] and, instead, focus on scaling to support large numbers of users.

Cloud computing can be defined as "a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers". Similar to we assume that the system is composed of the following parties: the Data Owner, many Data Consumers, many Cloud Servers, and a Third Party Auditor if necessary. To access data files shared by the data owner, Data Consumers, or *users* for brevity, download data files of their interest from Cloud Servers and then decrypt. Neither the data owner nor users will be always online. They come online just on the necessity basis. Cloud Servers are always online and operated by the Cloud Service Provider (CSP). They are assumed to have abundant storage capacity and computation power. The Third Party Auditor is also an online party which is used for auditing every file access event. In addition, we also assume that the data owner can not only store data files but also run his own code on Cloud Servers to manage his data files. This assumption coincides with the unified ontology of cloud computing which is recently proposed by Youseff et al.

More and more individuals are paying attention to the issue of privacy in cloud computing. As cloud services process user's data on machines that the users do not own or operate, this introduces privacy issues and can lessen user's control. Privacy issues are central to user concerns about adoption of cloud computing, and unless technological mechanisms to allay user's concerns are introduced, this may provide fatal to many different types of cloud services.For example, cloud service users report high levels of concern when presented with scenarios in which companies may put their data to uses of which they may not be aware. User's fears of leakage of commercially sensitive data and loss of data privacy may be justified: in 2007 the cloud service provider Salesforce.com sent a letter to a million subscriber describing how customer emails and addresses had been stolen by cybercriminals.

Some examples of emerging Cloud computing infrastructures are Microsoft Azure, Amazon EC2, Google App Engine, and Aneka.

Considering the above issues mentioned in section I, we propose the following solutions and put forth a new cloud architecture as mentioned in the figure 1.

1. Granting the access permission based on the user status.
2. Security while accessing the resource, by providing locking facility which will allow for the easy scale up and scale down of the resources.
3. Demand prediction methods to help the service providers in negotiating the Service Level Agreements(SLA).
4. Effective billing mechanism based on the usage. The ability to pay for use of computing resources on a short-term basis as needed and release them as needed, thereby rewarding conservation by letting machines and storage go when they are no longer useful.

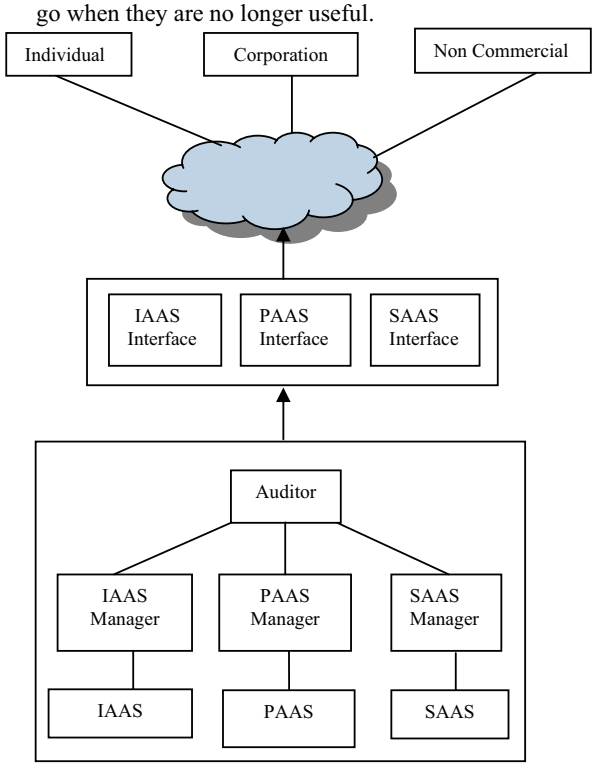

Figure 1.Secured, Managed and Scalable Cloud Architecture

3. SMS CLOUD

As mentioned in Fig 1, there are three basic types of cloud computing: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). In IaaS, cpu, grids or clusters, virtualized servers, memory, networks, storage and systems software are delivered as a service. Perhaps the best known example is Amazon's Elastic Compute Cloud (EC2) and Simple Storage Service (S3), but traditional IT vendors such as IBM, and telecoms providers such as AT&T and Verizon are also offering solutions. Services are typically charged by usage and can be

scaled dynamically, i.e. capacity can be increased or decreased more or less on demand.

3.1 Infrastructure as a Service.

IPs manage a large set of computing resources, such as storing and processing capacity. Through virtualization, they are able to split, assign and dynamically resize these re-sources to build ad-hoc systems as demanded by customers,
the SPs. They deploy the software stacks that run their services. This is the Infrastructure as a Service (IaaS) scenario.

3.2Platform as a Service

PaaS provides virtualized servers on which users can run applications, or develop new ones, without having to worry about maintaining the operating systems, server hardware, load balancing or computing capacity. Well known examples include Microsoft's Azure and Salesforce's Force.com. Microsoft Azure provides database and platform services starting at $0.12 per hour for compute infrastructure; $0.15 per gigabyte for storage; and $0.10 per 10,000 transactions. For SQL Azure, a cloud database, Microsoft is charging $9.99 for a Web Edition, which comprises up to a 1 gigabyte relational database; and $99.99 for a Business Edition, which holds up to a 10 gigabyte relational database. For .NET Services, a set of Web based developer tools for building cloud-based applications, Microsoft is charging $0.15 per 100,000 message operations.

3.3 Software as a Service

SaaS is software that is developed and hosted by the SaaS vendor and which the end user accesses over the Internet. Unlike traditional applications that users install on their computers or servers, SaaS software is owned by the vendor and runs on computers in the vendor's data center (or a collocation facility). Broadly speaking, all customers of a SaaS vendor use the same software: these are one-size-fits all solutions. Well known examples are Salesforce.com, Google's Gmail and Apps, instant messaging from AOL, Yahoo and Google, and Voice-over Internet Protocol (VoIP) from Vonage and Skype.

4. ADMISSION CONTROL IN SMS CLOUD

When a user wants to login in the SMS, he should first send the required role membership list associated with the cloud, the Administrator will generate the mapping roles according the collaboration policies and return to user a *ticket*.

- During the first step, the requester will send his identity and belonged role set(like commercial, non-commercial, etc) of its cloud to the Resource administrator.
- During the second step, the administrator will verify the validation of the user certification and the role set of its local cloud, and then assign the qualified roles (in a ticket with short validated time) to this user.
- During the last step, the user will send the service request to the target cloud.

4.1 Cloud Controller

The administrator act as a CLC. The CLC is a collection of web services which are best grouped by their roles into three categories:

- Resource Services perform system-wide arbitration of resource allocations, let users manipulate properties of the virtual machines and networks, and monitor both system components and virtual resources.
- Data Services govern persistent user and system data and provide for a configurable user environment for formulating resource allocation request properties.
- Interface Services present user-visible interfaces, handling authentication & protocol translation, and expose system management tools providing.

The Resource services process user virtual machine control requests and interact with the CCs to effect the allocation and deallocation of physical resources. A simple representation of the system's resource state (SRS) is maintained through communication with the CCs (as intermediates for interrogating the state of the NCs) and used in evaluating the realizability of user requests (**vis** a vis service-level agreements, or SLAs). The role of the SRS is executed in two stages: when user requests arrive, the information in the SRS is relied upon to make an admission control decision with respect to a user-specified service level expectation. VM creation, then, consists of reservation of the resources in the SRS, downstream request for VM creation, followed by commitment of the resources in the SRS on success, or rollback in case of errors.

The SRS then tracks the state of resource allocations and is the source of authority of changes to the properties of running reservations. SRS information is leveraged by a production rule system allowing for the formulation of an event-based SLA scheme. Application of an SLA is triggered by a corresponding event (e.g., network property changes, expiry of a timer) and can evaluate and modify the request (e.g., reject the request if it is unsatisfiable) or enact changes to the system state (e.g., time-limited allocations). While the system's representation in the SRS may not always reflect the actual resources, notably, the likelihood and nature of the inaccuracies can be quantified and considered when formulating and applying SLAs. Further, the admission control and the runtime SLA metrics work in conjunction to: ensure resources are not overcommitted and maintain a conservative view on resource availability to mitigate possibility of (service level) failures.

5.SECURITY USING LOCKS

The most widely used security on most ecommerce websites is the use of SSL. This secured socket layer encodes the information entered on your browser as it travels from the client to the server machine. This does ensure the protection for certain key data values such as passwords.

5.1 Access Control List

When the user logs onto a machine, it relies on the user account established on the operating system to control its access. In a similar way, you can add a layer of control to the access of your application. This will allow for controls to the specific functions and areas of your application that have greater granularity than the ones provided by the operating system. This can then be applied to functional roles that each user plays when using the system, such as an administrator would perform different tasks than a user. When accessing systems over the cloud, the user is removed from the operating system on the server. They are focused on getting the job done and not concerned with how their files are stored. It is therefore important for you to manage the user access separately from the operating system.

5.2 User and Data Access

The access control list allows you to identify and authenticate each user. The roles and user privileges granted to these individual users limit what access they have. This is important to control different functional areas of the system such as the distinction between administrators versus a casual user. Another important distinction is knowing which users will be granted access to what data. The databases that users access usually contain the business rules and consist of the primary intellectual property (IP) of your organization. It is therefore important to add an additional layer controlling users' access to the specific databases of the system.

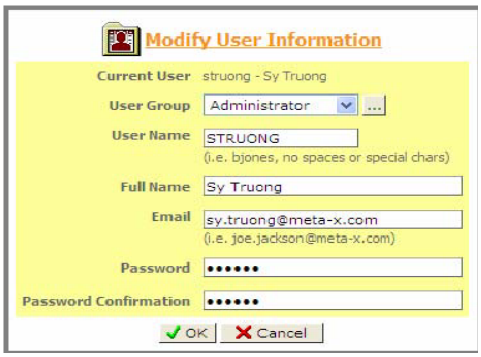

Figure 2.User Access is contributed by an account established for the application

6. DEMAND PREDICTION SCHEME

The usage of a Cloud client can sometimes have a repetitive behavior. This can be caused by the similarities between tasks that the Cloud client is running or the repetitive nature of human behavior. Given the self-similar nature of web traffic it follows that current usage patterns of online services have a probability of having already occurred in the past in a very similar form. Therefore we can infer what the system usage will be for a Cloud client by examining its past usage and extracting similar usages. The pattern strategy has two inputs: a set of past Cloud client usage traces and the present usage pattern that consists of the last usage measures of the Cloud client. Cloud clients working in the same application domain have a higher similarity in resource usages. Due to this similarity it follows that the most relevant historic resource usage data that can be used comes from Cloud clients working in the same application domain. Therefore it would make sense to isolate historical data based on application domains before usage. The present usage pattern of the Cloud client is used to identify a number of patterns in the historical set that are close to the present pattern itself. Identified patterns should not be dependent on their scale, just on the relation between the elements of the identified pattern and the pattern we are looking for. The resulting closest patterns will be interpolated by using a weighted interpolation (the found pattern that is closest to the present pattern will have a greater weight) and will have as result an approximation of the values that will follow after the present pattern. In essence, the usage of the Cloud client is predicted by finding similar usage patterns in the past or in other usage traces

7. BILLING SCHEME

7.1 Pay separately per resource

Most applications do not make equal use of computation, storage, and network bandwidth; some are CPU-bound, others network-bound, and so on, and may saturate one resource while underutilized Pay-as-you-go Cloud Computing can charge the application separately for each type of resource, reducing the waste of underutilization. While the exact savings depends on the application, suppose the CPU is only 50% utilized while the network is at capacity; then in a datacenter you are effectively paying for double the number of CPU cycles actually being used. So rather than saying it costs $2.56 to rent only $1 worth of CPU, it would be more accurate to say it costs $2.56 to rent $2 worth of CPU.

7.2Operations costs

Today, hardware operations costs are very low—rebooting servers is easy (e.g., IP addressable power strips, separate out of band controllers, and so on) and minimally trained staff can replace broken components at the rack or server level. On one hand, since Utility Computing uses virtual machines instead of physical machines, from the cloud user's point of view these tasks are shifted to

the cloud provider. On the other hand, depending on the level of virtualization, much of the software management costs may remain—upgrades, applying patches, and so on.

There are two additional benefits to the Cloud Computing user that result from being able to change their resource usage on the scale of hours rather than years.

- First, unexpectedly scaling down (disposing of temporarily underutilized equipment)—for example, due to a business slowdown, or ironically due to improved software efficiency— normally carries a financial penalty. With 3-year depreciation, a $2,100 server decommissioned after 1 year of operation represents a "penalty" of $1,400. Cloud Computing eliminates this penalty.
- Second, technology trends suggest that over the useful lifetime of some purchased equipment, hardware costs will fall and new hardware and software technologies will become available. Cloud providers, who already enjoy economy-of-scale buying power, can potentially pass on some of these savings to their customers. Indeed, heavy users of AWS saw storage costs fall 20% and networking costs fall 50% over the last 2.5 years, and the addition of nine new services or features to AWS over less than one year. 7 If new technologies or pricing plans become available to a cloud vendor, existing applications and customers can potentially benefit from them immediately, without incurring a capital expense. In less than two years, Amazon Web Services increased the number of different types of compute servers ("instances") from one to five, and in less than one year they added seven new infrastructure services and two new operational support options. Service providers can scale up or scale down their capacity.

8. CONCLUSION

Generally, Cloud Computing architecture constructed with reliable, secure, fault-tolerant, sustainable and scalable infrastructures for hosting Internet-based application services. Our SMS cloud architecture illustrates the on-demand self service for accessing the resources, providing security by using locking techniques, predicting the demand based on the client's resource utilization and based on the revenue of the cloud service provider, the billing scheme may vary.

9. REFERENCES

1. F. Berman, G. Fox, and T. Hey. Grid Computing: Making the Global Infrastructure a Reality. Wiley and Sons, 2003.
2. Salesforce Customer Relationships Management (CRM) system.
3. NSF TeraGrid Project. http://www.teragrid. org/.
4. T. Tannenbaum and M. Litzkow. The condor distributed processing system. Dr. Dobbs Journal, February 1995.
5. Foster, C. Kesselman, J. Nick, and S. Tuecke. The physiology of the grid: An open grid services architecture for distributed systems integration, 2002..
6. Foster, C. Kesselman, and S. Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. Int. J. High Perform. Comput. Appl., 15(3):200–222, 2001.
7. J. P. Ostriker and M. L. Norman. Cosmology of the early universe viewed through the new infrastructure. Communication. ACM, 40(11):84–94, 1997.
8. Amazon Web Services home page. http://aws. amazon.com

BIOGRAPHY

Prof.JabeenBegum.S received her M.E in Computer Science from Government College of Technology and is working towards Ph.D. in Computer Science from the Anna University of Technology, Coimbatore. Currently she is working as a HOD & Professor in CSE Dept, Velalar College of Engineering and Technology, Erode, Tamil Nadu, India and she is having 18 years of Experience in Teaching. She had published 21 Papers in Various National Conferences and she had presented 5 Papers in Various International Conferences held at many Engineering Colleges. Her Research Paper regarding "Time Complexity in Key Management" has published in AMSE, France, "An Effective Key Computation Protocol for Secure Group Communication in Heterogeneous Networks" published in IJCSNS and she had published her research paper in IEEE Explorer regarding Secure Group Communication. Her interests include Network Security, Distributed Systems, Cloud computing and DBMS.

Dr.T.Purusothaman is currently working as Associate Professor in the department of Computer Science and Engineering and Information technology, Government College of Technology, Coimbatore. He has twenty one years of teaching experience. He has completed Ph.D in the area of Network Security and Grid Computing. In his thesis, a novel key management scheme was proposed to provide service only for the paid customers in Internet. He has successfully completed a project funded by DIT (Government of India) in the area of cryptanalysis in the year 2006.

He has presented a number of papers in various National and International conferences. Many of his papers were published in IEEE Explore. He has to his credit several International Journal Publications in reputed journals including Journal of Grid Computing, Springer. His research interests include Network Security, Grid Computing and Data Mining.

G.Vidhya pursuing Final M.E. Computer Science &Engineering in Velalar College of Engineering and Technology, Erode, Tamil Nadu, India . She had published 4 Papers in Various National Conferences and she had presented one Paper in International Conference on Systemics, Cybernetics and Informatics under the aegis of Pentagram Research Centre, Hyderabad, India on 5th jan 2008. She has undergone her project regarding "An Efficient Deterministic Key Pre-Distribution Scheme for Multiple Attacks". In her project she concentrates on favorable degree of resilience in deterministic key pre-distribution scheme. Her interests include Network Security, DBMS and Cloud computing.

R.Ebenezer Princy pursuing Final M.E. Computer Science &Engineering in Velalar College of Engineering and Technology, Erode, Tamil Nadu, India . She had published 3 Papers in Various National Conferences held at many Engineering Colleges. She has undergone her project regarding "Preserving Data Integrity in Cloud Computing". Her interests include Information Security and Cloud computing.