

July 2012

ENHANCEMENT FOR DATA SECURITY IN CLOUD COMPUTING ENVIRONMENT

GAWALI M. B.

1,2Computer Engineering Dept., RCPIT , Shirpur, India, gawali_mahen@yahoo.com

R. B. WAGH

Computer Engineering Dept., RCPIT , Shirpur, India,, raj_wagh@rediffmail.com

S. P. PATIL

Computer Engineering Dept. GECA, Aurangabad, India, shivajipatil999@yahoo.com

Follow this and additional works at: <https://www.interscience.in/ijcct>

Recommended Citation

B., GAWALI M.; WAGH, R. B.; and PATIL, S. P. (2012) "ENHANCEMENT FOR DATA SECURITY IN CLOUD COMPUTING ENVIRONMENT," *International Journal of Computer and Communication Technology*. Vol. 3 : Iss. 3 , Article 9.

DOI: 10.47893/IJCCT.2012.1139

Available at: <https://www.interscience.in/ijcct/vol3/iss3/9>

This Article is brought to you for free and open access by the Interscience Journals at Interscience Research Network. It has been accepted for inclusion in International Journal of Computer and Communication Technology by an authorized editor of Interscience Research Network. For more information, please contact sritampatnaik@gmail.com.

ENHANCEMENT FOR DATA SECURITY IN CLOUD COMPUTING ENVIRONMENT

GAWALI M. B.¹, R. B. WAGH² & S. P. PATIL³

^{1,2}Computer Engineering Dept., RCPIT, Shirpur, India, ³Computer Engineering Dept. GECA, Aurangabad, India
E-mail : gawali_mahen@yahoo.com¹, raj_wagh@rediffmail.com², shivajipatil999@yahoo.com³

Abstract - Cloud computing, a rapidly developing information technology, has aroused the concern of the whole world. Cloud computing is Internet-based computing, whereby shared resources, software and information, are provided to computers and devices on-demand, like the electricity grid. Cloud computing is the product of the fusion of traditional computing technology and network technology like grid computing, distributed computing parallel computing and so on. It aims to construct a perfect system with powerful computing capability through a large number of relatively low-cost computing entity, and using the advanced business model like SaaS (Software as a Service) to distribute the powerful computing capacity to end users' hands. To address this longstanding limitation by building a multi-tenant system. Our system provides the environment for the user to perform his tasks, but with very high security. By using further facilities provided in this system user will be secure about his data and his account.

Keywords - PaaS, IaaS, SaaS, Multi – tenant, MaaS etc.

I. INTRODUCTION

A. The Cloud Computing

The term *cloud* has been used historically as a metaphor for the Internet. This usage was originally derived from its common depiction in network diagrams as an outline of a cloud, used to represent the transport of data across carrier backbones (which owned the cloud) to an endpoint location on the other side of the cloud. This concept dates back as early as 1961, when Professor John McCarthy suggested that computer time-sharing technology might lead to a future where computing power and even specific applications might be sold through a utility-type business model. This idea became very popular in the late 1960s, but by the mid-1970s the idea faded away when it became clear that the IT-related technologies of the day were unable to sustain such a futuristic computing model. However, since the turn of the millennium, the concept has been revitalized. It was during this time of revitalization that the term *cloud computing* began to emerge in technology circles [1].

B. The Emergence of Cloud Computing

Utility computing can be defined as the provision of computational and storage resources as a metered service, similar to those provided by a traditional public utility company. This, of course, is not a new idea. This form of computing is growing in popularity, however, as companies have begun to extend the model to a cloud computing paradigm providing virtual servers that IT departments and users can access on demand. Early enterprise adopters used utility computing mainly for non-mission-critical needs, but that is quickly changing as trust and reliability issues are resolved. Some people think cloud computing is the next big thing in the world of IT. Others believe it is just another variation of the utility computing model that has been repackaged in

this decade as something new and cool. However, it is not just the buzzword “cloud computing” that is causing confusion among the masses. Currently, with so few cloud computing vendors actually practicing this form of technology and also almost every analyst from every research organization in the country defining the term differently, the meaning of the term has become very nebulous. Even among those who think they understand it, definitions vary, and most of those definitions are hazy at best. However, when “the cloud” is combined with “computing,” it causes a lot of confusion. Market research analysts and technology vendors alike tend to define cloud computing very narrowly, as a new type of utility computing that basically uses virtual servers that have been made available to third parties via the Internet. Others tend to define the term using a very broad, all-encompassing application of the virtual computing platform. They contend that anything beyond the firewall perimeter is in the cloud. A more tempered view of cloud computing considers it the delivery of computational resources from a location other than the one from which you are computing.

C. SaaS Multi Tenant-

The architecture of SaaS-based applications is specifically designed to support many concurrent users (multi tenancy) at once. This is a big difference from the traditional client/server or application service provider (ASP)- based solutions that cater to a contained audience. SaaS providers, on the other hand, leverage enormous economies of scale in the deployment, management, support, and maintenance of their offerings. Multi-tenancy is an organizational approach for SaaS applications. Although SaaS is primarily perceived as a business model, its introduction has led to numerous interesting problems and research in software engineering. Despite the growing body of research in this area, multi-tenancy is still relatively unexplored, despite

the fact the concept of multitenancy first came to light around 2005. While a number of definitions of a multi-tenant application exist, they remain quite vague. Therefore, we define a multi-tenant application as the following:

Definition 1. A multi-tenant application lets customers (tenants) share the same hardware resources, by offering them one shared application and database instance, while allowing them to configure the application to fit their needs as if it runs on a dedicated environment.

Definition 2. A tenant is the organizational entity which rents a multi-tenant SaaS solution. Typically, a tenant groups a number of users, which are the stakeholders in the organization. These definitions focus on what we believe to be the key aspects of multi-tenancy:

1. The ability of the application to share hardware resources.
2. The offering of a high degree of configurability of the software.
3. The architectural approach in which the tenants (or users) make use of a single application and database instance[2].

II. RELATED WORK

The cloud is a next generation platform that provides dynamic resource pools, virtualization, and high availability. Today, we have the ability to utilize scalable, distributed computing environments within the confines of the Internet, a practice known as cloud computing[3]. Cloud Computing is becoming a well-known buzzword nowadays. Many companies, such as Amazon, Google, Microsoft and so on, accelerate their paces in developing Cloud Computing systems and enhancing their services to provide for a larger amount of users. However, security and privacy issues present a strong barrier for users to adapt into Cloud Computing systems. In this paper, we investigate several Cloud Computing system providers about their concerns on security and privacy issues. We find those concerns are not adequate and more should be added in terms of five aspects (i.e., availability, confidentiality, data integrity, control, audit) for security. Moreover, released acts on privacy are out of date to protect users' private information in the new environment (i.e., Cloud Computing system environment) since they are no longer applicable to the new relationship between users and providers, which contains three parties (i.e., Cloud service user, Cloud service provider/Cloud user, Cloud provider). Multi located data storage and services (i.e., applications) in the Cloud make privacy issues even worse. Hence, adapting released acts for new scenarios in the Cloud, it will result in more users to step into Cloud. We claim that the prosperity in Cloud Computing

literature is to be coming after those security and privacy issues having be resolved[4].

There is a critical need to securely store, manage, share and analyze massive amounts of complex (e.g., semi-structured and unstructured) data to determine patterns and trends in order to improve the quality of healthcare, better safeguard the nation and explore alternative energy. Because of the critical nature of the applications, it is important that clouds be secure. The major security challenge with clouds is that the owner of the data may not have control of where the data is placed. This is because if one wants to exploit the benefits of using cloud computing, one must also utilize the resource allocation and scheduling provided by clouds. Therefore, we need to safeguard the data in the midst of untrusted processes[5].

Cloud computing, a rapidly developing information technology, has aroused the concern of the whole world. Cloud computing is Internet-based computing, whereby shared resources, software and information, are provided to computers and devices on-demand, like the electricity grid. Cloud computing is the product of the fusion of traditional computing technology and network technology like grid computing, distributed computing parallel computing and so on. It aims to construct a perfect system with powerful computing capability through a large number of relatively low-cost computing entity, and using the advanced business models like SaaS (Software as a Service) [6]. With the development of parallel computing, distributed computing, grid computing, a new computing model appeared. The concept of computing comes from grid, public computing and SaaS. It is a new method that shares basic framework. The basic principles of cloud computing is to make the computing be assigned in a great number of distributed computers, rather than local computer or remoter server. The running of the enterprise's data center is just like Internet. This makes the enterprise use the resource in the application that is needed, and access computer and storage system according to the requirement. This article introduces the background and principle of cloud computing, the character, style and actuality. This article also introduces the application field the merit of cloud computing, such as, it do not need user's high level equipment, so it reduces the user's cost. It provides secure and dependable data storage center, so user needn't do the awful things such storing data and killing virus, this kind of task can be done by professionals. It can realize data share through different equipments. It analyses some questions and hidden troubles, and puts forward some solutions, and discusses the future of cloud computing. Cloud computing is a computing style that provide power referenced with IT as a service. Users can enjoy the service even he knows nothing about the technology of cloud computing and the

professional knowledge in this field and the power to control it[7].

Existing systems –

A. *The Force.com*

Introduction –

In 2008, The Force.com provides the first Multitenant architecture for SaaS. The focus of this system is multi tenancy, a fundamental design approach that can dramatically help improve the manageability of SaaS applications. This figure 1 defines multi tenancy, explains the benefits of multi tenancy, and demonstrates why metadata driven architectures are the premier choice for implementing multi tenancy. The world’s first PaaS, which delivers turnkey multi tenancy for Internet-scale applications. The system details Force.com’s patented metadata-driven architecture components to provide an understanding of the features used to deliver reliable, secure, and scalable multitenant applications.

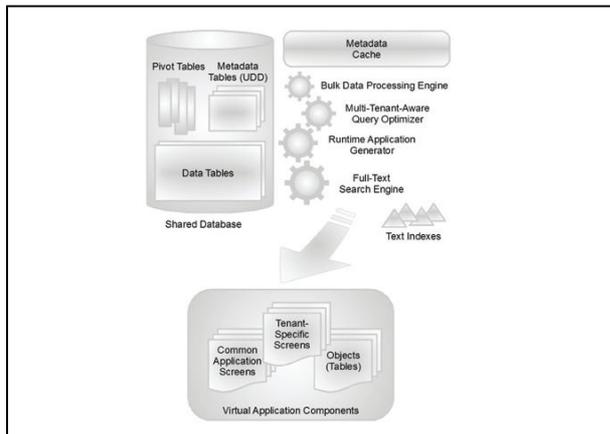


Fig. 1 : Force.com Platform Architecture.

Forms, reports, work flows, user access privileges, tenant-specific customizations and business logic, even the definitions of underlying data tables and indexes, are all abstract constructs that exist merely as metadata in Force.com’s Universal Data Dictionary (UDD). For example, when a developer is building a new custom application and defines a custom table, lays out a form, or writes some procedural code, Force.com does not create an “actual” table in a database or compile any code. Instead, Force.com simply stores metadata that the platform’s engine can use to generate the “virtual” application components at runtime. When someone wants to modify or customize something about the application, all that’s required is a simple non-blocking update to the corresponding metadata.

Because metadata is a key ingredient of Force.com applications, the platform’s runtime engine must optimize access to metadata; otherwise, frequent metadata access would prevent the platform from scaling. With this potential bottleneck in mind, Force.com uses metadata caches to maintain the

most recently used metadata in memory, avoid performance sapping disk I/O and code recompilations, and improve application response times. Force.com stores the application data for all virtual tables in a few large database tables that serve as heap storage. The platform’s engine then materializes virtual table data at runtime by considering corresponding metadata.

To optimize access to data in the system’s large tables, Force.com’s engine relies on a set of specialized pivot tables that maintain denormalized data for various purposes such as indexing, uniqueness, relationships, etc. Force.com’s data processing engine helps streamline the overhead of large data loads and online transaction processing applications by transparently performing data modification operations in bulk. The engine has built-in fault recovery mechanisms that automatically retry bulk save operations after factoring out records that cause errors[8].

System Strength :

1. This is first SaaS multi tenant system.
2. System focused on most of the database related terms.

System Lack :

1. Any detailed information is not provided about event log.
2. No information given about the user profiles.
3. No mapping is provided.
4. System again lacking on tenant management.
5. No any specific algorithm is given for Customer account security.

B. *LABS^{hp}*

Introduction –

On 21 Feb 2009, LABS^{hp} provides Multi-tenancy in Cloud – based Collaboration Services.

In this system it is mentioned that increased outsourcing of non-core competencies will drive the demand for a new generation of multi-tenanted cloud-based platforms that address the needs of content-centered collaboration between organizations. This system is based on the FRACTAL conceptual prototype which has allowed us to evaluate the suitability of current enterprise content management (ECM) technologies for this type of platform.

Fractal Conceptual Prototype –

Goals of the Prototype

We had three distinct goals for the prototype: first, we wanted a functioning system that would help us to better envision FRACTAL from an end user perspective; second, we wanted to clarify requirements for the underlying platform; and third, we wanted to understand limitations of current ECM technologies for realizing multi-tenanted cloud-based applications.

Key Features –

Content Spaces : hosted spaces that bring together people, content, collaborative tools, and customizable active behaviors.

Active behaviors : a way for end users to define functional extensions operating within the context of a content space involving content, metadata, automated processing services and tasks carried out by other users. An active behavior may be manually invoked as needed, or it may be automatically triggered by a change to a content space or the passing of time. An invocation may involve a single content object or many objects in parallel. Their complexity ranges from automatically creating up to date PDF versions of documents as they are modified, to running workflows to automatically collate information from several collaborating organizations into a single document.

Agile configuration : must be light-weight, low-touch and customizable by end users without IT involvement.

Open and extensible by third parties : an Internet platform with open APIs, where third parties are motivated to develop customizations/extensions that can then be published through a marketplace and easily discovered by end users[9].

System Strengths :

1. System explain prototype for Multi-tenant.
2. System properly handled the SaaS multi-tenant concept.

System Lack :

1. System lack on users security.
2. System lack on event log.
3. System lack on mapping.
4. System lack on Load distribution.

C. EMC²

Introduction-

In June 2009, EMC² provides powerful capabilities for creating Software as a Service(SaaS). This system offers SaaS built on top of the Documentum 6.5 platform stack. The primary focus is on multi-tenancy as one of the major areas of complexity.

This system provides multi tenancy with following repository

1. *Shared repository* – model isolation is achieved via partitioning the data within single repository. The main reason for choosing a shared repository model is to support efficiently a large amount of small customers with a very load. If an application serves customers with one to 10 users it is probably the best model. Typically, a unique Customer ID is associated with the data that must be partitioned. All queries must be qualified by Customer ID to guarantee that customer will never have access to others customers data.

2. *Dedicated repository* – In this case, each customer will be guaranteed isolation from all possible perspectives. Using the dedicated model minimize the impact on application design and implementation. The only requirement is a mapping of the Customer ID to a repository. In shared repository data isolations are also provided.

3. *Content Isolation* - The simplest way to isolate the content is by creating a base custom type and making all content-enabled types derived from it.

User Isolation- In shared repository take care that each username does not collide with the names of other uses. This is important since the username is actually a UID and is used by other objects to refer to a user.

3.1 *Schema Isolation* - Two types of schema elements are most common : Custom attribute and Custom values. In Custom attribute are often required to model extension to document type such as Contacts or purchase orders. Each customer could have a unique set of metadata necessary for each type of document. In custom values one would need to create a special type and manage all the customer-defined values separately[10].

System Strengths :

1. This system focused on the user profile.
2. By using the data isolations the customer have the facilities to work more efficiently.
3. Customers account is safe because of the above mentioned repository.

System Lacks :

1. System more focused on customer instead the SaaS multi tenancy.
2. No method given through that user get more details about applications.
3. No work schedule is mentioned.
4. The mapping is not provided to the customer.
5. No security norms are mentioned.

- ✓ Security
- ✓ Privacy
- ✓ Reliability
- ✓ Compliance
- ✓ Freedom

III. PROPOSED SOLUTION

As on the based on the above mentioned ‘Literature Survey’ we try to give proposed solution which will covered probably all lacunas of the above described systems. Multi-tenancy refers to the ability to run multiple customers on a single software instance installed on multiple servers. This is done to increase resource utilization by allowing load balancing among tenants, and to reduce operational complexity and cost in managing the software to deliver the service.

Three Tier Architecture of Web Services

As shown in following fig. 2, the developed solutions are divided into following layers.

Presentation Layer

The presentation layer provides an interface for clients to access the portal application. This layer consists of the following elements:

- ✓ **Web Forms:** The primary web form is the Default.aspx. This page is the entry point to the portal. It is responsible for loading the other elements of the presentation layer.
- ✓ **Skins:** The Default.aspx web form loads the skin for the page based on the settings for each page or portal.
- ✓ **Containers:** The Default.aspx web form also loads the containers for the modules based on the settings for each module, page, and portal.
- ✓ **Module User Controls:** Modules will have at least a single user control that is the user interface for the module.

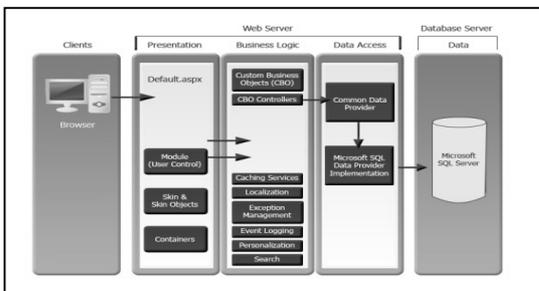


Fig. 2 : Three Tier architecture of Web Services

Business Logic Layer

The business logic layer provides the business logic for all core portal activity. This layer exposes

many services to core modules. These services include

- ✓ Localization
- ✓ Caching
- ✓ Exception Management
- ✓ Event Logging
- ✓ Personalization
- ✓ Search
- ✓ Installation & Upgrades
- ✓ Security

Data Access Layer

The data access layer consists of two elements:

- ✓ **Data Provider API:** This is an abstract base class that establishes the contract that the implementation of the API must fulfil.
- ✓ **Implementation of Data Provider API:** This class inherits from the Data Provider API class and fulfills the contract by overriding the necessary members and methods. The core SaaS Multi-Tenant release provides a Microsoft SQL Server implementation of the Data Provider API.

Data Layer

The data layer provides data to the data access layer. The data store used in the data layer must be supported by the implementation of the Data Provider API to fulfil the data requests.

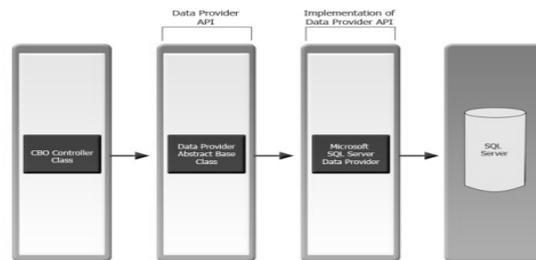


Fig. 3 : Data layers

Securities

To secure the data in cloud computing, we try to proposed Digital Signature with RSA encryption algorithm[2].

So, we are proposed to do the following.

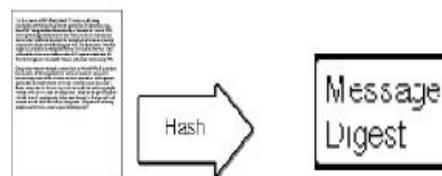


Fig. 4 : Document crunch into message Digest



Fig. 5 : Encryption of message digest signature

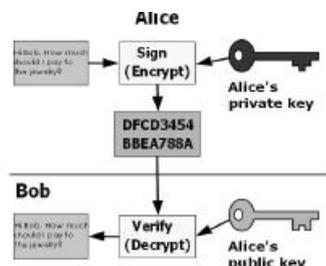


Fig. 6 : Encryption of Digital Signature into cipher text

With the increase of the amount of data and users in information system, the requirements of data integrity in system need to be improved as well. To handle the multiple users in cloud computing again proposed MD5 algorithm[11] which is shown in the figure 6.

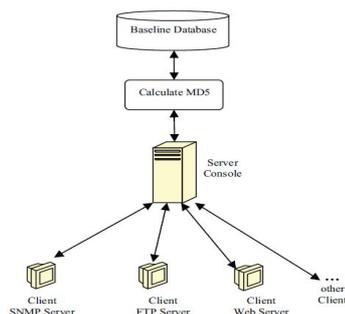


Fig. 7 : System architecture of MD5

IV. CONCLUSIONS

Cloud computing is still struggling in its accuracy, with positive and negative comments made on its possible implementation for a large-sized enterprise. IT technicians are spearheading the challenges. Several groups have recently been formed, with the goal of exploring the possibilities offered by cloud computing and to establish a common language among different providers. In this cloud computing is facing several issues in gaining recognition for its merits. Its security deficiencies and benefits need to be carefully handle before making a decision to implement it.

After study of various IT companie’s SaaS Multi-tenant system, one thing is clear that, it is very fast growing technology now days. But, each SaaS multi-tenant have some remedies which make it less powerful for working. This proposed system will

overcome on most of the remedies in previous systems.

ACKNOWLEDGEMENT

At the time of making a survey on this area many peoples were helped me. I specially thanks Prof. R. B. Wagh and Prof.G.K. Patnaik and for his valuable guidance on this area.

REFERENCES

- [1] John W. Rittinghouse, James F. Ransome, “Introduction”, Cloud Computing- Implementation, management & security, Ed. Taylor & Francis Group, LLC ,2010, pp.27-29.
- [2] Cor-Paul Bezemer, Andy Zaidman, “Multi-Tenant SaaS Applications: Maintenance Dream or Nightmare?”, Delft University of Technology Software Engineering Research Group Technical Report Series, 2010, <http://www.se.ewi.tidelft.nl/techreports>, pp 1-5.
- [3] Uma Somani, Kanika Lakhani, Manish Mundra, “Implementing Digital Signature with RSA Encryption Algorithm to Enhance the Data Security of Cloud in Cloud Computing”, 2010 1st International Conference on Parallel, Distributed and Grid Computing (PDGC - 2010), pp.211-216.
- [4] Minqi Zhou, Rong Zhang, Wei Xie, Weining Qian, Aoying Zhou, “Security and Privacy in Cloud Computing: A Survey”, 2010 Sixth International Conference on Semantics, Knowledge and Grids, pp. 105-112.
- [5] Kevin Hamlen, Murat Kantarcioglu, Latifur Khan, Bhavani Thuraisingham, “Security Issues for Cloud Computing”, International Journal of Information Security and Privacy, 4(2), April-June 2010, pp 39-51.
- [6] Jianfeng Yang, Zhibin Chen, “Cloud Computing Research and Security Issues”, International Conference on Computational Intelligence and Software Engineering(CiSE), Wuhan, 10-12 Dec. 2010, pp 1-3.
- [7] Shuai Zhang, Shufen Zhang, “Cloud Computing Research and Development Trend”, 2010 Second International Conference on Future Networks,ICFN’10, 2010, pp 93-97.
- [8] The Force.com Multitenant Architecture: Understanding the Design of Salesforce.com’s Internet Application Development Platform. http://www.salesforce.com/au/assets/Force.com_Multitenancy_WP_101508.pdf.
- [9] David Banks, John S. Erickson, Michael Rhodes, “Multi-tenancy in Cloud-based Collaboration Services”, Hewlett-Packard Development Company, L.P., February 21, 2009 <http://www.hpl.hp.com/techreports/2009/HPL-2009-17.pdf>.
- [10] “Implementing SaaS Multi-tenancy with EMC Documentum 6.5- Best Pratices Planning”, EMC2, <http://www.emc.com/collateral/software/white-papers/h4701-oem-multitenancy-wp.pdf>.
- [11] Danyang Cao, Bingru Yang, “Design and implementation for MD5-based data integrity checking system”, International Conference on Information management and Engineering (ICIME), 2010, pp.608-611.

