Automating Resource Management of Sub Servers in Cloud Computing Environment

M.Kumudha[1], Dr. M. Arunachalam[2]
PG Scholar[1], Prof. and HOD/ CSE [2]
[1,2,]Department of Computer Science and Engineering
[1,2,]KLN College of Information Technology,Pottapalayam, Tamilnadu, India

Abstract

Cloud computing is an emerging data interactive paradigm to realize user's data remotely stored in an online cloud server. Cloud services provide great conveniences for the users to enjoy the on-demand cloud applications without considering the local infrastructure limitations. At the time of data accessing, different users may be in a collaborative relationship, and thus data sharing becomes significant to achieve productive benefits. The existing system focuses more on Threat reduction which has the possibility to get reengineered. Security is the major concern and Load balancing is also one of the dominant issues in Cloud computing environment. The system is designed such that it bridges the gap between Security and Quality of Service. Many of the touted gains in the cloud model come from resource multiplexing through virtualization technology.

The system uses virtualization technology to allocate data center resources dynamically based on application demands and supports green computing by optimizing the number of servers in use. The Resource Management is automated by "Ultrasonic protocol". A set of heuristics are developed that prevent overload in the system effectively while saving energy used. Multifactor Authentication (MFA) is implemented that allows Instance Number Generation and Modification of OTP's to enhance protection towards resource allocation such that these mechanisms are difficult to reengineer. At the same time VM's of one machine can be operated in parallel from the other Machine. Trace driven simulation and experiment results demonstrate that the algorithm achieves good performance.

Index terms: *Multifactor Authentication, OAuth Token, Instantly generated Token, Skewness*

1. Introduction

Cloud being on-demand, is very flexible, more reliable and also elastic. The Data resting in the cloud needs to be accessible only by those who are authorized to do so. Security directives are key challenges and more important, but more difficult to achieve. The traditional methods of managing security aren't scaling to the growth of the threat landscape.

Apart from security, there are several other factors such as load balancing and resource management which has to be concentrated more for providing data Integrity and timely delivery of data is also an essential factor that has to be achieved. It is required for any system to balance the load for speeding up the response and to reduce the work of the server. Some mechanisms are required to reduce the manual work and automate the entire process for speedy execution.

2. Objectives

This system uses virtualization technology to allocate data center resources dynamically based on application demands and support green computing by optimizing the number of servers in use. The system introduces the concept of "skewness" to measure the unevenness in the multidimensional resource utilization of a server. It combines different types of workloads and improves the overall utilization of server resources. The system also develops a set of heuristics that prevent overload in the system effectively while saving energy used. Thus overload avoidance, Green Computing, Automating Resource management, handling multiple request in very high speed and to achieve a security for the above that can never be reengineered are the key objectives of the system.

3. Module Description

- Multifactor Authentication
- Task Assignment
- User Request Analysis
- Server Load value
- Server Allocation

3.1 Multilevel oAuth Authentication

A new secure token integrated with a cyclic periodic random number apart from traditional password system which progress to a next level of authentication which checks for the oAuth code. The system Initiates a three level of authentication.

A First level

A Secured registered password. This is the initial level of authentication which is the usual procedure of providing the User Login Id and the Password that is been created during Registration process.

B Second level - (4 digit registered token + instantly generated token)

During Registration the user will be generated a Unique Token which is of four digits. The server generates a random number which will be prevailing only for specific time period, at the end of the time period the existing token will disappear and the system will generate the other set of numbers (Token) which is more of random cycles. The Instance for the generated token will be running on the server. The system checks for authentication where a mixture of inputs are been provided to the system for logging in successfully.

1st 10 seconds = 4 digit Registered Token + Instance Number1

2nd 10 Seconds = 4 digit Registered Token + Instance Number2 … etc

The system now checks whether the 1st four numbers are the mapped registered token and the next 6 numbers are the server generated user number

Eg: A54g+189901

Polynomial based Secure Algorithm generates Secret key at particular Time intervals which is more of cyclic manner and generates other Key for next Cycle

C Third level OAuth - (Open Authentication→4 digit registered token + OTP) token

OAuth Token is the combinations of both Unique Id and the OTP that is been generated during the authentication process. An oAuth token which was formed by hashing once generated OTP with mitigated password.

Step1: OTP is given as Input; the system modifies the OTP and return the result to the User

Step 2: OTP along with the registered token is given in combination as mentioned below

OTP = AD08DE
Hashed OTP = *!$5u0
Combined Input : oAuth token = *! + (SecretKey) + $5 + (SecretKey) + u0

Polynomial based password hashing algorithm hashes the once generated password in a secure manner. After successful login the system will redirect to the customer information screen.

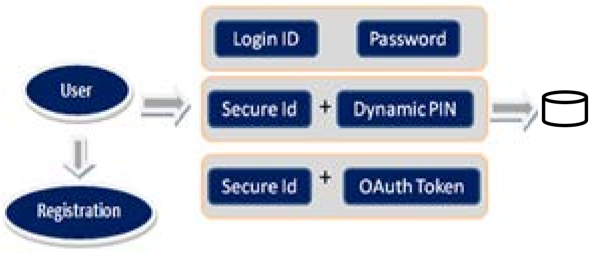

Fig 1: Multifactor Authentication

3.2 Task

The module deals with the user requesting to download the required file. The request will be stored and processed by the server to respond the user. It checks the appropriate sub server which is nothing but the Virtual Machines to assign the task. A Job scheduler is a computer application for controlling unattended background program execution; Job scheduler is created and connected with all servers to perform the user requested tasks. The Mapping of Server data to the Sub-servers located as layers is been performed. Task is been performed by Virtual Machines (VM's) and the control is been provides by Physical Machines (PM's).

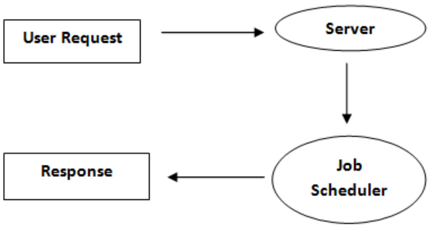

Fig 2: Task Assignment

3.3 User Request Analysis

The users input requests are analyzed by the scheduler before the task is given to the servers. This module helps to avoid the task overloading by analyzing the nature of the users request. First it checks the type of the file going to be downloaded. The users are allowed to select the file out of their interest and the system recognizes the file type, displays all the files based on the type of file requested. The users request can be the downloading request of text files (document, PDF, etc), image (JPEG, PNG, etc) or video file (MP4, AVI).

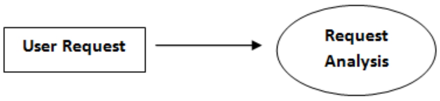

Fig 3: User Request Analysis

3.4 Server Load value

In this module, the server load value is identified for job allocation. To reduce the over load, the different load values are assigned to the server according to the type of the processing file. If the requested file is text, then the minimum load value will be assigned by the server. If it is video file, the server will assign high load value. If it is image file, then it will take medium load value. Thus the task is to assign values for the files based on its capacity.

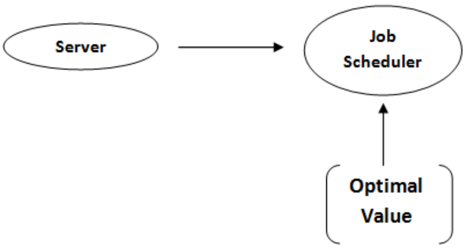

Fig 4: Server Load Value

3.5 Server Allocation

In this module, the server allocation task will take place. To moderately manage mixed workloads, the Job scheduling algorithm is followed. In this scheduling, depends upon the nature of the request the load values are assigned dynamically. Minimum load value server will take high load value job for the next time. High load value server will take minimum load value job for next time. The prioritization of task based on the file type is been performed.

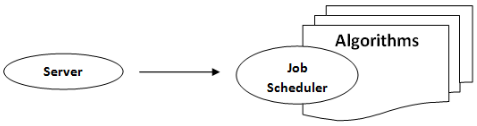

Fig 5: Server Allocation

4. System Design

The system initiates with registration phase following with layers of security checks. It continues with hiding the automation process with sheepdog mechanism. Design is multi-step process that focuses on Security, data structure software architecture, procedural details, (algorithms etc.) and interface between modules. The design process also translates the requirements into the presentation of software that can be accessed for quality before Processing begins.

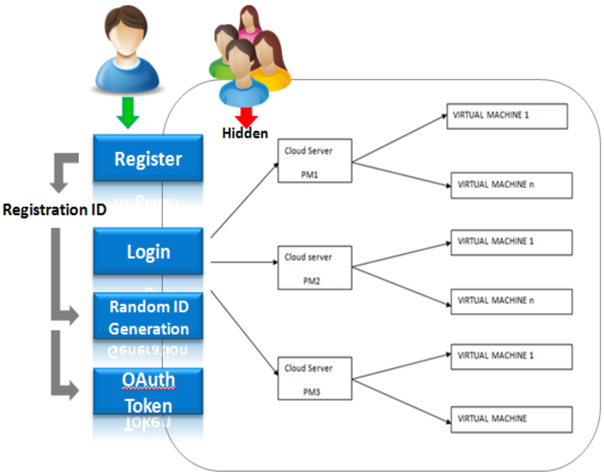

Fig 6: Architecture

5. Proposed Work

The system proposes an attractive model for multi-user collaborative cloud applications. Several layers of VM's are implemented such that Server1 →VM1, VM2…VMn; Server2 → VM1, VM2…VMn; Server3 → VM1, VM2…VMn. Greedy Algorithm is used for Configuring, developing and setting up of layers of Virtual Machines. Mapping of virtual machines (VMs) to physical resources is been performed such that all the VM's are evaluated for Resource allocation.

After completion of each successive VM's the system recognizes the nearby VM's for assigning the task which is done in parallel. Gossip protocol is used for mapping Resources to VM's and Ultrasonic Protocol is used to identify the nearby nodes. Mapping is largely hidden from the cloud users. Sheepdog mechanism hides VM's. Traffic Redundancy Elimination Algorithm activates client and servers, enables VM's for handling the data in a systematic way & Fine Grained Algorithm is responsible for proper orientation of the data and handles the acknowledgment task for the user request. Multifactor Authentication provides security across several layers. The system presents the design and implementation of an automated resource management system.

5 6. Conclusion and Future Work

Cloud-based service-oriented applications have the potential to self-adapt their QoS, depending on demand. This is a market-based mechanism that maps the real-world situation of unpredictable change of QoS requirements, costs involved in adaptation and adaptation by competing applications. As the number of possible concrete services increases, the scalability of the self-adaptive mechanism and time scale up becomes important.

Security on the other hand is also a critical parameter. This market-based mechanism consisting of simple agents is able to adapt well and yet scales linearly to the number of concrete services. It is also robust in the presence of differences in demand and supply of QoS. It also bridges the gap between security and Quality of Service. The future work of the current system is that it may be even better to implement all these mechanisms using Platform machine which is the advancement of the Virtual Machine and it is a widely evolving Technology to handle all sort of processing.

References

[1] Niharika Gupta, Rama Rani: "Implementing High Grade Security in Cloud using Multifactor Authentication and Cryptography", International Journal of Web & Semantic Technology (IJWesT) Vol.6, No.2, April 2015

[2] Sumathi M, Sharvani G.S, Dinesha H A: "Implementation of Multi Factor Authentication System For Accessing Cloud Service", International Journal of Scientific and Research Publications, Volume 3, Issue 6, June 2013

[3] Deepa Panse, P. Haritha: "Multi-Factor Authentication In Cloud Computing For Data

Storage Security", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 8, August 2014

[4] T.Lakshmi Praveena, V.Ramachandran, CH. Rupa: "Attribute based Multifactor Authentication for Cloud Applications", International Journal of Computer Applications (0975 – 8887)Volume 80 – No 17, October 2013

[5] P. Mell and T. Grance, "Draft NIST Working Definition of Cloud Computing," Nat'l Inst. of Standards and Technology, 2009.

[6] A. Mishra, R. Jain, and A. Durresi, "Cloud Computing: Networking and Communication Challenges," IEEE Comm. Magazine, vol. 50, no. 9, pp. 24-25, Sept. 2012.

[7] R. Moreno-Vozmediano, R.S. Montero, and I.M. Llorente, "Key Challenges in Cloud Computing to Enable the Future Internet of Services,"IEEEInternet Computing,vol.17,no. 4,pp. 18-25July/Aug.2013.

[8] K. Hwang and D. Li, "Trusted Cloud Computing with Secure Resources and Data Coloring," IEEE Internet Computing, vol. 14, no. 5, pp. 14-22, Sept./Oct. 2010.

[9] J. Chen, Y. Wang, and X. Wang, "On-Demand Security Architecture for Cloud Computing," Computer, vol. 45, no. 7, pp. 73-78, 2012.

[10] Y. Zhu, H. Hu, G. Ahn, and M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231-2244, Dec. 2012.

[11] H. Wang, "Proxy Provable Data Possession in Public Clouds," IEEE Trans. Services Computing, vol. 6, no. 4, pp. 551-559, Oct.-Dec. 2012

[12] K. Yang and X. Jia, "An Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing," IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 9, pp. 1717-1726, Sept. 2013.