EFFICIENT SERVER USING AUTOMATED VM GENERATION

R.Divya[1] | P.Kamali[2]
[1](Dept of Computer Science, Anand Institute of Higher Technology, Chennai, India, divyaravichandran2008@gmail.com)
[2](Department of Computer Science, Anand Institute of Higher Technology, Chennai, India, kamalikama96@gmail.com)

Abstract— *In the current scenario most of the service providing companies provide services through server-client manner. Here important focus is on how the server is responding to the request put forth by the client. The efficiency of the server is in how fast it is providing the required information to the client and also performing computation in short time if required. Most of the servers deals with millions of users where it works with the help of the cloud where multiple clients gets response at the same time. Cloud computing is the process of using internet to store, processing and working with files from remote host where it provides huge storage support. The server that is serving multiple users makes use of this cloud when needed. Serving users at the same time without server slowdown is major problem. The challenging issue is to improve the server response which provides services to all the client request at the same time . Allocating service capacities in cloud computing is based on the assumption that they are unlimited and can be used at any time. The proposed system is the automatic creation of virtual machine when the server is about to get overloaded. The virtual machine remains active for the given timeslot and gets destroyed automatically when the server comes out of the peak hour. An iterated heuristics framework is presented for the problem under study which mainly consists of initial solution construction, improvement strategies are proposed.*

Keywords—Cloud Computing; Server; Virtual Machine

1. INTRODUCTION

Cloud computing is an area where we can store the resources. It plays a major role in providing space to the resources where users can access resources at any time through the internet. Multiple requests can be provided to the cloud simultaneously. It manages the workload by allocating the multiple request among the servers. A Server is designed in a way to provide information based on the requests. Request is often put forth by the Client (User). As all the server deals with the Clients it is often referred as Client-Server model. Single server can be used for many processes. Some of the services provided by the Server are information or data sharing among multiple clients and it also performs computations for the Clients(User).

The Server and the Client can be available within the same device or different device which are connected over the network. Now-a-days most of the enterprises uses the cloud based servers. Whenever the Client places the request ,the Server listens to them and then provides the required information. A Server acts as a center of processing request. There are many existing issues in the Server that deals with multiple requests. Server overloading is one of the central issue in cloud related servers. A Server should be available to all the users at any time. Multiple requests should not slowdown the server efficiency.

When the Server is handling different tasks(for ex: transaction, shopping, payments) at same Server by allocating certain space to the task, if all the tasks are served at the same time server response will slow down so to avoid this server will set a peak time to particular task with maximum request and minimum for other tasks .By this server will serve based on user request and satisfy user end.

The aim of this paper is to avoid the server overloading so that the user can easily access the resources they need, reduction in startup time and to minimize the cost of the cloud users. Multiple clients send request to the server at a same time, when the capacity of the server exceeds it cannot handle the incoming request. So that the client has to wait for some time to access the resources, which can be managed by the automated virtualization technology. The incoming request is handled simultaneously therefore the user can access the resources whenever they need without any delay.

2. RELATED WORKS

[1] Truthful Greedy mechanisms for dynamic virtual machine provisioning and allocation in clouds. The dynamic VM and allocation in cloud problem is designed by the truthful greedy mechanism. The G-VMPAC-X truthful greedy mechanism collects the request from the user and determines the allocation of resources by calling the allocation algorithm. The near optimal solution is determined by the proposed greedy mechanism which effectively captures the dynamic market demand and it generates high revenue.

[2] Network aware scheduling for virtual machine workloads with interference model. Consolidation of resource intensives virtual machine and variance in workload performance is solved by introducing an extensive benchmark suite and profiling framework. The network aware scheduling for VM workloads with interference model identify and reduce network related interference effects using performance models based on the runtime characteristics of virtualized workloads.

[3] Striking a balance between traffic engineering and energy efficiency in virtual machine placement. It investigate how traffic engineering(TE) and energy efficiency(EE) goals in virtual machine placement can

coexist with the emergence of virtual bridging and of multipath forwarding . A versatile formulation of VM placement problem provides support to the virtual bridging and multipath forwarding, proposing a repeated matching heuristic for its solution.

3. PROPOSED SYSTEM

All the user who is trying to retrieve information from the server face the problem of server crash and much cost spent in the maintenance of large storage area without any loss of data.

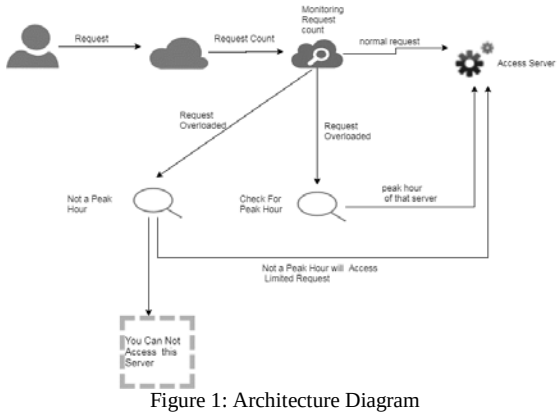

Figure 1: Architecture Diagram

In the above fig 1: the clear explanation of how the system is going to be is explained. The user is the one who asks for the information which is kept stored in the server. Certain mechanisms are used to retrieve the information from the server. There are various types of server. Peak hour is the time set by the admin who notices in which time what work is carried out. If the server limit is not exceeded then the request will be handled by the main server and if the limit is exceeded then the virtual machine is generated automatically in the cloud. If server is overloaded in the not peak hour then the user will never get any response from the server.

The information is provided to the user with the help of virtual machine which does the same work as the server but less efficient than main server. In the proposed system we have to first find the server peak hour based on the count recorded by the system, later the system will be designed in order to provide information to any number of user within that peak hour.

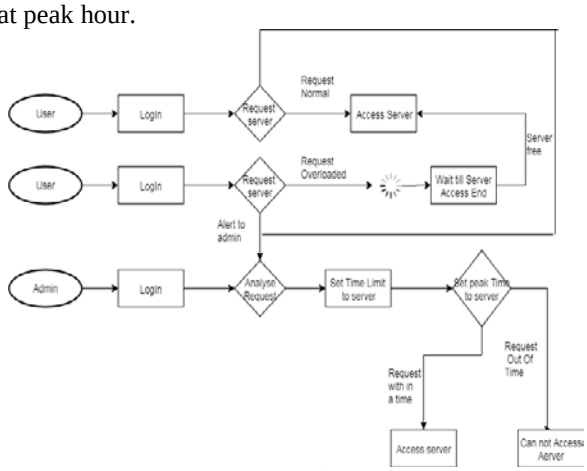

Figure 2: Flow Diagram for accessing server

If it is not the peak hour the request will be responded by the main server itself. From fig 2: It is clearly explained how the user and the server work together to retrieve the information.

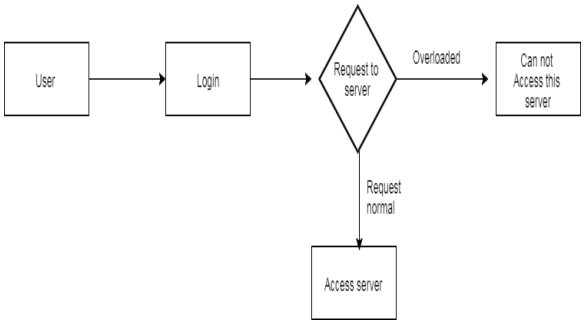

Figure 3: Flow Diagram for User Login

Fig 3: The user will first login as the entry to the page to get the required information from the server. After selecting which type of work and server the user will send the request. The system will check whether it is the peak hour or not. If the request is sent during not peak hour the response will be provided by the normal server and when the request is made during the peak and when the server limit exceeds the response will be provided by the virtual server. The admin can also able to login in with the help of username and password to view the work performed by the user, to set the time limit and to view the count.

Server consolidation is implemented with the help of time slot filtering. Fig 4: It generates the request by the user. It also monitor the available server to check whether the server limit exceeds or not. If the limit is reached the virtual machine is created but if it does not exceed then the server itself will handle the request. After all the works are carried out the response will be provided to the user.

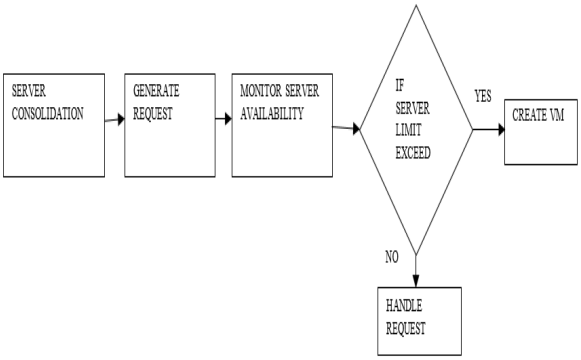

Figure 4: Server Consolidation

4. IMPLEMENTATION

The implementation of the efficient server using automated VM generation is developed by using HTML,CSS,JAVASCRIPT as the front end tool while JAVA,PHP,MY SQL as the back end tool. JDBC is used to provide the database connectivity. PHP is used to support the server side scripting language.

Figure 5: Home page

Home page helps the user, admin to login in to the system for their further proceedings. New user can register in to the system by using the registration page.

Figure 6: Diagram of workflow

A. USER

This system allows the user to create an account and fill the necessary details which will be stored in the database such that the user need not have to do the whole registration process every time. Then the user can login to the system whenever they need to do transaction, shopping, to view result.

B. ADMIN

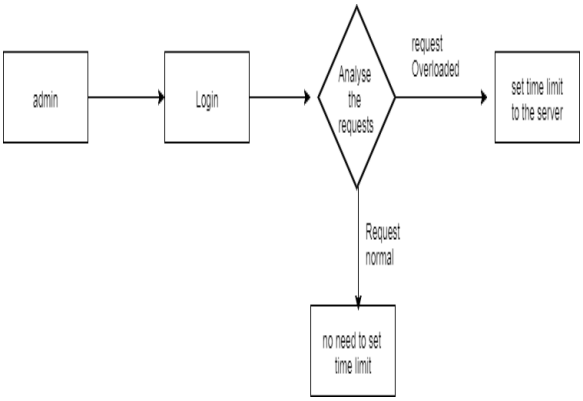

Figure 6.1: Admin Flow Diagram

Admin can able to view the user's request, the work of setting time limit to the server is carried out and also the admin will view the total number of request count. If there is any change needed in the system it is made by the admin. He has to log in with the login id and password. Admin store the details of the user in the database for future use. Fig 4: Admin can analysis the user request based on particular duration. Then admin can decide at which time which particular server busy admin can make at that respective time server as busy and other server with minimum number of request. Before setting a time limit to server admin has to analyze request at lest of one week. Based on the user request to the server on timings then admin can able to decide to set a which particular time server will be peak hour so has to access more number of request without getting server slowdown. And rest of the timing server will act normal which mean will access only minimum number of request.

5. CONCLUSION AND FUTUREWORK

The work related to server is continuous and emerging platform delivering services to the users in many manners. There are many mass social information providing websites working with the help of this server. As these servers are providing information to huge number of users it should be efficient. This paper deals with the way of providing information to the users without any delay. Peak hour is set and only during this time the virtual machine is generated to provide the response. The vision of providing the admin to view the number of users online and what type of work is carried out by them. The admin can also able to set the time limit to the server and in this time the server can serve any number of user with the automatic generation of virtual machine.

The future enhancement is adding features to the proposed idea of this paper. We designed this in a way where only in the server peak time it will provide response to any number of users but it can be improved in a way that the software designs itself after the analysis of the work carried out with the help of the server and in which time it is carried out. In the available system we are using different servers to different work but it can be changed into one server doing multiple tasks in the future.

REFERENCES

[1] Mahyar Nejad, Daniel Grosu, Lena Mashayekhy, "Truthful Greedy Mechanisms for Dynamic Virtual Machine Provisioning and Allocation in Clouds", IEEE Transactions on Parallel and Distributed Systems, Volume:26, Issue2: Publication Pages:594-603 ,2015.

[2] Sam Verboven, Kurt Vanmechelen and Jan Broeckhove, "Network Aware Scheduling for Virtual Machine Workloads With Interference model", IEEE Transaction on Services for Computing, Volume: 8, Issue: 4, Publication Pages: 617-629, 2015.

[3] Dallal Belabed , S.Secci , Guy Pujolle , Deep Medhi, " Striking a Balance Between Traffic Engineering and Energy Efficiency in Virtual Machine Placement", IEEE Transactions on Network and Service Management, Volume: 12, Issue: 5, Publication Pages: 202-216, 2015.

[4] S. Yang, P. Wieder, and R. Yahyapour, "Reliable virtual machine placement in distributed clouds", in Proc. Of 8th IEEE/IFIP International Workshop on Reliable Networks Design and Modelling (RNDM), 2016, PP. 1-7.

[5] M. Menzel, R. Ranjan, L. Wang, S. Khan, and J. Chen, "Cloud genius: A hybrid decision support method for automating the migration of web application clusters to public clouds," IEEE Transactions on Computers, vol. 64, no. 5, Publication Pages: 1336–1348, 2015.

[6] M. Alicherry and T. Lakshman, "Network aware resource allocation in distributed clouds", in Proc. of IEEE INFOCOM, Publication Pages: 963-971, 2012.

[7] A. G. Delavar and Y. Aryan, "Hsga: a hybrid heuristic algorithm for workflow scheduling in cloud systems," Cluster computing, vol. 17, no. 1, Publication Pages: 129–137, 2014.

[8] Amir Varasteh , Maziar Goudarzi, "Server Consolidation Techniques in Virtualized Data Centers: A Survey", IEEE Systems Journal, vol.11, no. 2 , publication pages: 772-783, 2017.

[9] Hyang-Won Lee, Eytan Modiano, Kayi Lee, Member, "Diverse Routing in Networks With Probabilistic Failures", IEEE/ACM Transactions on Networking, Volume: 18, Issue: 6, Publication Pages: 1895-1907, 2010.

[10] Qingya She, Xiaodong Huang, "How Reliable Can Two-Path Protection Be", IEEE/ACM Transactions on Networking, Volume: 18, Issue: 3, Publication Pages: 922-933, 2010.