Evaluation of Cloud Hybrid Load Balancer (CHLB)

Po-HueiLiang[1], Jiann-Min Yang[2]

Department of Management Information Systems, National ChengChi University, Taipei, Taiwan

[1]lph@iii.org.tw; [2]jmyang@mis.nccu.edu.tw

***Abstract*-With the lower cost of service maintenanceand the easieravailability of cloud computing services, users have increasinglymoved their Web resourcesto the cloud environment. Consequently, users can access their resources with the browsers of their thin devices, and cloud service providers do not need to buy numerous machines for the uncertain requirements of backup or expansion. The cloud servicescan also be dynamically providedfor customers, so the users pay as they use computing resources for their requirements. This technology drives manyinnovativetypes of services. The availability and performance of the client systems will be the next main concerns in cloud computing.**

This paper presents a framework for load balancing of the Web sites in a cloud with round robin Domain Name System(RRDNS). The proposed framework,called thecloud hybrid load balancer (CHLB), is intended for adapting an open-source load-balancing system with RRDNS technology for a specific Web cluster. This CHLB framework allows the network service provider to deploy a software-based load balancer dynamically while the customers need the load balancer for their hosted Web clusters.

Keywords-Cloud Computing; Load Balance; Round Robin DNS

I. INTRODUCTION

Cloud computing provides the capabilities to offer different applications over the Internet as services. This enablesWeb applications in which the resources are dynamically scalable and often virtualized [4, 15]. Many international application providers such as Amazon, Google, Microsoft, and Internet datacenter (IDC) have begun to establish new types of IDC for hosting cloud computingapplicationstoachieve redundancy and reliability to meet their service level agreements. Since user requirements for cloud services are varied, service providers must ensure that they can deliver their services flexibly in a cloud environment [4].

The purpose of this paper is to present an open-source solution to build a hybrid load-balancing architecture that is scalable for Web clusters in cloud environments. The rest of this paper is organized as follows: Section II gives an overview of related studies on load balancing. Section IIIdescribes the features and advantages of cloud computing. Section IV presents the framework of thecloud hybridloadbalancer (CHLB) and lists the test results of the proposed framework. Finally, Section V discussesthe conclusions and future research work.

II. RELATED STUDIES ON LOAD BALANCING

The most common problem for a Webservice to resolve is the traffic problem caused by the large number of online users on the peak time. For a Web administrator, the major improvement problemsare to enhance Web performance and to provide advanced services for users. Many load-balancing solutions have been proposed and subsequently used to overcome these issues, including commercial and open-sources[26]. To solve various load problems, the known load-balancing algorithms can be divided into two major categories: static [1,5-9]and dynamic [1-3,10-12]. The scheduling of the static algorithm is carried out according to a predetermined approach,whilethe dynamic algorithm adapts its decision to the current state of the system. Therefore, the dynamic approach is more responsive to changes in system parameters. The choice of load-balancing algorithms is a difficult task. Most of the algorithms that have been proposed, are based on specific application requirements and system environments.

Based onthe type of the platform, the load balancers can be divided into the hardware based and the software based. The hardware-based solutions adapt the physical resources, such as network switches, to establish the load-balancing environment.There are some very powerful products providing the load balance functionalities, such as A10 network AX 2500 load balancer for the hardware-based network load balancer [27] and the Citrix NetScaler for both the hardware and software based network load balancer [28]. However, the costs by deploying these solutions to build the scalable web services are very expensive.

RRDNS is a technique of load distribution, load balancing, and fault-tolerance provisioning for redundant Internet Protocol service hosts,whereby the DNSresponses to the address requested from client computers managed according to an appropriate statistical model [29]. In the simplest implementation, RRDNS respond to the DNS requests bylisting the IP addresses for several servers that host identical services. The order in which IP addresses from the list are returned is the basis for the term round robin [2, 29, 31].Preliminary work on the distribution and assignment of incoming connections [32-34] has relied on RRDNS to distribute the incoming connections across a cluster of servers. Due to intricacies of the DNS protocol, RRDNS was found to have limited value for the purposes of load balancing and fault tolerance with scalable Web server clusters[32, 33].

These load balancing techniques canspread the load of Internet systems, and such applications have been important network tools in the service environment. However, these load balancers might break down or the service bandwidth might become fully loaded. Hence, the most important work is that done to failover the service. The following are some issues that should be considered [30]:

- What is the cost to build the highly available services?
- How many staffsneed to support the new architecture?
- Is there any economical approach for solving the problem?

In this paper, we propose an economical load-balancing framework with cloud computing techniques. The enterprises can save the expenseof purchasing new machines and hiring more IT staff.

III. OVERVIEW OF CLOUD COMPUTING

Cloud computing is a way to deliver services over the network, and it advances with virtualization technology. Cloud computing can provide the ability to add capacity as needed, typically with very short lead times. Based on virtualization technology, cloud computing provides new types of on-demand IT services and products [21]. Several common features of cloud computingaresummarized by the following:1) on-demand service, 2) ubiquitous network access, 3) location-independent resource pooling, 4) rapid elasticity, and 5) pay per use [16-20].

To realize cloud services, the fundamental step is to virtualize resources such as the hardware, network, and applications [21], as shown in Fig. 1. Thus, anIDC can dynamically "provision" resources on demand [22, 23]. The three common types of basiccloud services, described in Table 1, are calledinfrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS) [16, 22-24, 35]. Based on the virtualization of the physical resources from IaaS, the more extensive types of services listed in Fig. 2 are created.

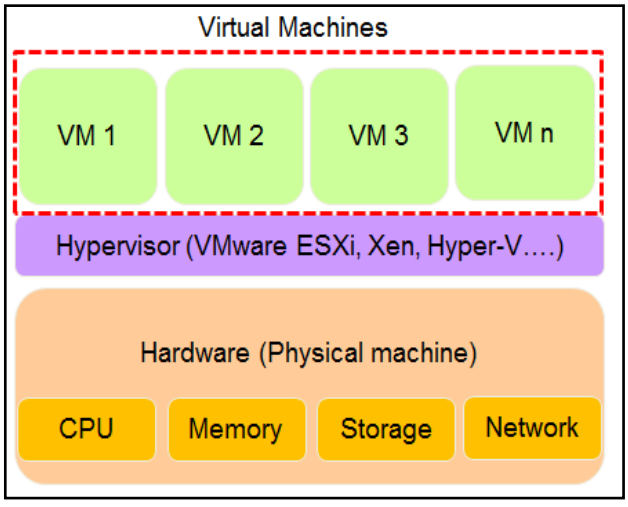

Fig.1Virtualization of a physical machine

TABLE I THREE BASIC CLOUD SERVICES

Type	Description
Infrastructure as a Service (IaaS)	• It is a service of deploying the virtualized hardware resources. • This service provides raw virtual machine instances, storage, and computation at pay-as-you-go utility pricing.
Platform as a Service (PaaS)	• To provide theapplicationplatform for internetprogramming interface, operation platform,andetc. • The services step up from pure utility computing are Cloud platform services, which hide virtual machine instances behind higher-level APIs.
Software as a Service (SaaS)	• The software service on Internet can beusedthrough standard Web browser. • Any web application is a Cloud application service in the sense that it resides in the Cloud.

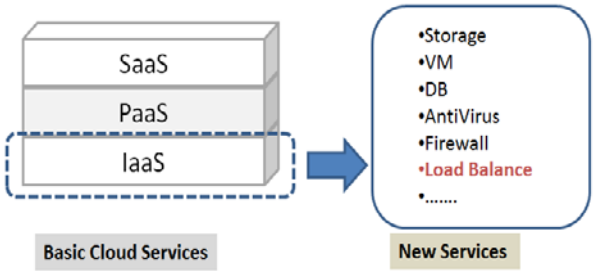

Fig. 2 Newservices from IaaS

On the other hand, the application scope of cloud services includes the Internet, hardware, system software, operation management, and other related resources [14]. To reach the goals of a cloud service, the virtualization technology should be adapted for "provisioning" the required resources dynamically [20]. Thus, the operators of the cloud data centers provide the established cloud resources with virtualized nodes instead of physical nodes [22, 23]. Moreover, the quality assurance of cloud services is crucial to ensuring that the service level agreement is upheld. To deliver the guaranteed quality of services in a cloud computing environment, it is necessary to incorporate other management tools, such as a network firewall and a load balancer. Based on the resource consolidation of cloud computing, the user workloads and the management tools can be executed on a random virtual machine (VM). Additionally, the lower costs of the hardware virtualization force IT staff to choose the virtualized solutions [13, 23, 30].

IV. CLOUDHYBRID LOAD BALANCER

With the advantages of the lower cost and the scalability from Cloud Computing,we implement the CHLB in the cloud environment. The three main components of the CHLB include the RRDNS, the load balancing system and the web system. Each component can be one or groups of virtual machines.To share the efforts of the load balancer and to avoid the main DNS service fail, at least two of the

RRDNS VMs include all web IPs information and those RRDNS IP must be registered to the global DNS service provider.The responsibility of the load balancing system is to receive the http requests and thenredirect them to the web system. It could be a single VM or a cluster for the high availability purpose. The web system receives the requests from the load balancing system, and transfers the data to the users. If some VMs need to be closed for the system maintenance purpose, the new alternative VMs can be deployed through requests. In the framework, users do not need prepare any hardware machines, network environments and the IT staffs.

A simple CHLB environment is illustrated in Fig. 3, the load balancer VM and the RRDNS VMarethe key components for a Web VM cluster. We adapt the load balancer to the Linux Virtual Server (LVS)[26] for the specific Web cluster in our framework. The LVS can provide several techniques, such as network address translation, direct routing, and IP encapsulation, to distribute IP packets among nodes. Our proposed system chooses IP tunneling, which allows the LVS administrator to put servers on different network segments, enabling our system to scale upfor more clusters and nodes in the cloud environment. In the CHLB, the LVS is a virtual machine in the cloud and is supportedonly for one specific Web cluster in the same hypervisor.

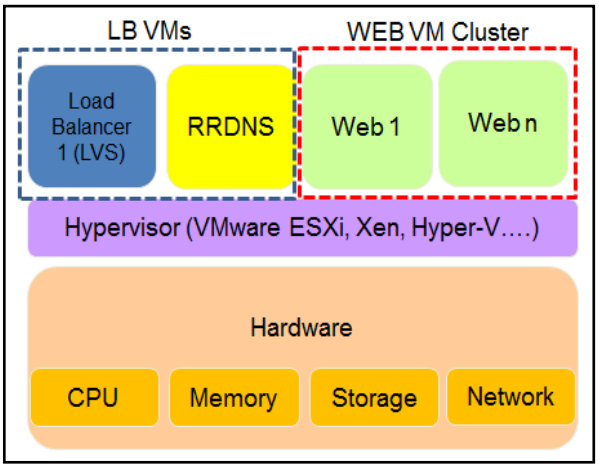

Fig. 3 Proposed architecture of thecloud hybrid load balancer

The cloudhybrid load balancer (CHLB) proposed in this paper is an open-sourcehybridload-balancing solution for Web clusters. This framework can be applied with other kinds of hypervisors, which can deploy new LVS VM instances, as shown in Fig. 4. The LVS VM is responsible for balancing the loads imposed by the traffic of a group of Web servers. The RRDNS arranges the HTTP requests from the users and transfers the linksto the different LVS addresses. Then, the LVS redirects to the real Web server,which returns the HTTP results to the user. The purpose of this research was to develop an open-source solution that can rapidly be reused in the cloud environment,since the costs of these virtual load balancers are much less than those of the customary physical load balancers. Furthermore, the CHLB can decrease the activity of the LVSbymeans of the RRDNS.

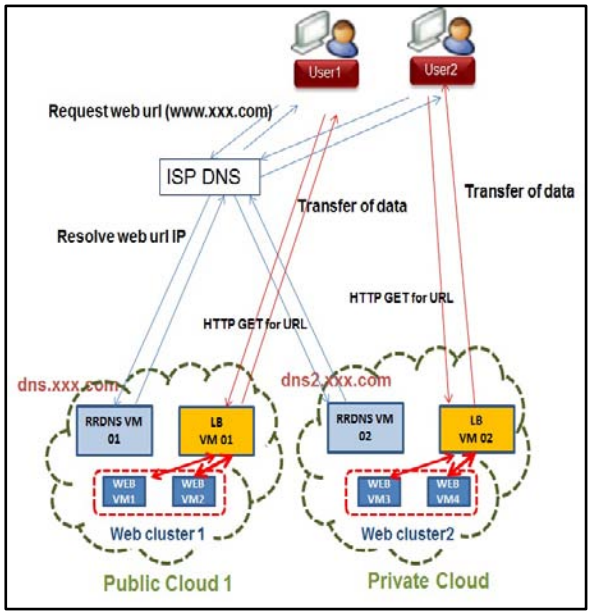

Fig. 4The framework of the CHLB

A. Architecture of CHLB

Fig. 4 shows the architecture of the proposed cloudhybrid load balancer (CHLB) in the cloud environment. This framework combines with the Webclustersand the load-balancing VMs in the hybrid cloud environment. Each Web cluster includes its own network load balancer (LVS),and the LVSsystems are setup for high availability of a specific Web service. The RRDNS VM is responsible for sequentially arranging the destination IP address of the Web server cluster, and this function can spread the load of the first Web cluster. However, the second RRDNS VM becomes the primary DNS. Based on the characteristic of the global DNS registry service, the second DNS can decrease the failover time when the first DNS breaks down.

The information process can be described as the Fig. 4. When the “ISP DNS” receives the http request from the User1, the ISP DNS would check its DNS registry database and find the registered DNS and then transfers the http requests to the “RRDNS VM01”. The “RRDNS VM01” would resolve the IP information and send to User1. Therefore the User1 would get the right IP to connect the load balancing system “LB VM01”, the “LB VM01” would redirect the request to the “WEB VM1”. Finally, the User1 would get web data. Because we register both the IP information of the “LB VM01” and the “LB VM02” in the “RRDNS VM01” and the “RRDNS VM02”for the same domain name, the http request of next user “User2” should be returned from “LB VM02”. On the other hand, we register two DNS information in the “ISP DNS”, the “RRDNS VM02”would be responsible for the resolution of the http request while the service of “RRDNS VM01” is crashed.

In order to save on the cost of Web resources and IT staff, the proposed system is builtin the cloud

environment.While the users can subscribe to more resources, the performance of the Web cluster slows down, or another Web environment can be establishedin another cloud. This kind of Web cluster would be scalable with the cloud IaaS.

B. Implementation and Evaluation of CHLB

The proposed system was established and then evaluated. In the experimental environment, weadaptedeight virtual machines created for the CHLB.The first evaluation is to verify the high availability of the RRDNS. For the testing, we stop the major RRDNS VM "RRDNS VM01". We found that the http request can be resolved by the second RRDNS VM "RRDNS VM02" and return the web data correctly. The other evaluation is to validate the load balancing functionalities of the proposed framework. We make the different home pages on the four different web VMs, and then we use two isolated machines to connect the web system. The results of the two machines are different and this shows that our proposed framework is effective.

The next demonstration explores the test results of the Webvirtual machines in threedifferent scenarios: 1)fourWeb VMs with anLVS load balancer, 2) fourWeb VMs with anRRDNSloadbalancer, and3)fourWeb VMs with hybrid LVS and RRDNS loadbalancers. In our experiment, we use twoHP ProLiantDL385G6 servers, each containing onesix-core 2.4-GHz AMD CPU with hyper-threading support and running the hypervisor of the CitrixXenServer[28]. Each serveris configured with 16 GB of memory and is connectedwith a single 10-Mbit Internet bandwidth. The hardware configuration of allWebVMs isone vCPU and 1024 MB of RAM. The software-based load balancer is built with the Linux Virtual Server systemonaCentOS Linux VM, and its configuration is one vCPU and 1024 MB of RAM. The RRDNS VM is established with one vCPU and 4096 MBofRAM. The DNS environment is adopted with Windows 2008 DNS.

The load-generating client adopted isa Toshiba R10 notebookwith the Microsoft Windows 7 Professionaloperating systeminstalled. The hardware configuration of the client is one two-core CPU and 4096MB of RAM. The Web stress tool used on the client is Pylot1.26,which is an open-source Webperformance testing tool[25]. Each testing agent is run with the same configuration, which includes the duration of 60 s, the ramp-up time of 0 s, and the interval time of 0 ms.The stress testing is conducted with various agent parameters and the test results are gathered in TableII. The results show that the requests generated by the CHLB were much more the results of the other two mechanisms.

TABLE IITESTRESULTS FORWEB PERFORMANCE

Testing Cases	Testing Agent #	Testing results				
		Data received (bytes)	Total Requests	Avg Resp time(secs)	Throughput (req/sec)	errors
(1)Four web VMs	250	406,730	2,285	0.824	190.417	0
	500	106,444	598	0.347	149.500	0
with LVS	750	83,126	467	0.344	155.667	0
	1,000	202,082	1,279	1.130	142.111	47
(2)Four web VMs with RRDNS	250	228,468	1,446	0.824	180.750	0
	500	89,902	569	0.346	142.250	0
	750	87,058	551	0.353	183.667	0
	1,000	181,542	1149	0.912	143.625	0
(3)Four web VMs with CHLB	250	243,162	1,539	0.773	192.375	0
	500	1,795,328	11,338	2.541	149.184	2
	750	226,414	1,433	1.055	179.125	1
	1,000	1,622508	10,178	4.914	111.846	2

V. CONCLUSIONS AND FUTURE WORK

This paper has demonstrated the applicability of CHLB techniques to obtain improvements in the resource utilization and availability of a cloudcomputing environment. Based on the features of the cloud services, theproposed approach offers major cost advantages to cloud vendors.

Thus far, we have discussed 1) the basic concepts of cloud computing and load-balancing techniques, 2) theproposed load-balancing technique that is based on the CHLB,and3) how the virtualized LVS can balance the load of a Webclusterbyusing the IP tunneling method. Moreover, it has been proved that the hybrid load-balancing framework, which is combined with the LVS and the RRDNS, can decrease the load of the LVS virtual machines and can dynamically adjust the weights of the real Web servers.

In this paper, we have proposed a framework withhybridload-balancing to ensure the scalability and availability in cloud computing environments. Future work isto achieve this and to extendthe framework to ensure the reliability of the mobile agents involved.The VM-based Web cluster is scheduled by the LVS, and the Web server implementation uses Apache server. This framework of the CHLBcan be easily reused in other commercial hypervisor environments and the cloud IaaS. This contribution motivates IT staffs to save the costs of switch-type network load balancers.

Although the CHLBconcept has been proved, many aspects need to be improved and compared with those of other load-balancing algorithms. Our future work will focus on how to improve the performance of the CHLB for large users over the Internet and make comparisons with other kinds of load-balancing algorithms.

ACKNOWLEDGMENT

This study is conducted under the "Core Technology Research and Development for Cloud Application Project" of the Institute for Information Industry, which is subsidized by the Ministry of Economy Affairs of the Republic of China.

REFERENCES

[1] R. Tong and X. Zhu,"A Load Balancing Strategy Based on the Combination of Static and Dynamic," in Database Technology and Applications (DBTA), 2010 2nd International Workshop, pp. 1-4, 2010.

[2] Cardellini V, Colajanni M, Yu PS, "Dynamic load balancing on web-server systems," IEEE Internet Comput 3(3):28–39, 1999.

[3] Dhakal, S., Hayat, M.M., Pezoa, J.E., Yang, C.,Bader, D.A., "Dynamic load balancing in distributed systems in the presence of delays: a regeneration theory approach," IEEE Trans. Parallel Distrib, Syst.18(4), 485–497, 2007.

[4] Dobber, M., Koole, G., Mei, R., "Dynamic load balancing experiments in a Grid," In: Proceedings of IEEE International Symposium on Cluster Computing and the Grid, Cardiff, 2005.

[5] ChristofWeinhardt, ArunAnandasivam, Benjamin Blau, JochenStosser, "Business Models in the Service World," IT Professional, pp.28-33, 2009.

[6] Kameda, H., Li, J., Kim, C., Zhang, Y., "Optimal Load Balancing in Distributed Computer Systems," Springer, London, 1997.

[7] Tang, X., Chanson, S.T., "Optimizing static job scheduling in a network of heterogeneous computers,"In: Proceedings of Intl. Conf. on Parallel Processing, pp. 373–382. IEEE, Piscataway, 2000.

[8] Grosu, D., Chronopoulos, A.T., "Non cooperative load balancing in distributed systems," J. Parallel Distrib.Comput.65(9), 1022–1034, 2005.

[9] Penmatsa, S., Chronopoulos, A.T., "Job allocation schemes in computational Grids based on cost optimization," In: Proceedings of 19th IEEE International Parallel and Distributed Processing Symposium, Denver, 2005.

[10] Penmatsa, S., Chronopoulos, A.T., "Price-based user optimal job allocation scheme for Grid systems," In: Proceedings of 20th IEEE International Parallel and Distributed Processing Symposium, Rhodes, 2006.

[11] Penmatsa, S., Chronopoulos,A.T., "Dynamicmulti-user load balancing in distributed systems," In: Proceedings of 21st IEEE International Parallel and Distributed Processing Symposium, Long Beach, 2007.

[12] Shah, R., Veeravalli, B., Misra, M., "On the design of adaptive and de-centralized load balancing algorithms with load estimation for computational Grid environments," IEEE Trans. Parallel Distrib.Syst.18, 1675–1686, 2007.

[13] Arora, M., Das, S.K., Biswas, R., "A de-centralized scheduling and load balancing algorithm for heterogeneous Grid environments," In: Proceedings of International Conference on Parallel Processing Workshops, pp. 499–505. IEEE, Piscataway, 2002.

[14] B. P. Rimal, E. Choi, and I. Lumb, "A taxonomy and survey of cloud computing systems," in NCM '09: Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC. Washington, DC, USA: IEEE Computer Society, pp. 44–51, 2009.

[15] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia., "Above the Clouds: A Berkeley View of Cloud computing," Technical Report No. UCB/EECS-2009-28, University of California at Berkley, USA, Feb. 10, 2009.

[16] Hutchinson, C.; Ward, J.; Castilon, K., "Navigating the Next-Generation Application Architecture," IT Professional, vol.11, no.2, pp.18-22, 2009.

[17] D. S. Linthicum, "Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide," Addison-Wesley Information Technology Series, Addison Wesley, 2009.

[18] M. B. Greer, "Software as a Service Inflection Point: Using Cloud Computing to Achieve Business Agility,"iUniverse, Inc., 2009.

[19] Menken, and G..Blokdijk, "Cloud Computing Virtualization Specialist Complete Certification Kit - Study Guide Book and Online Course,"Emereo Pty Ltd, 2009.

[20] T. Velte, Cloud Computing: A Practical Approach, McGraw-Hill, USA, 2009.

[21] Rittinghouse, Cloud Computing: Implementation, Management, and Security. 1st edition, CRC Press, 2009, 26. M. Vouk: Cloud Computing—Issues, Research, and Implementations. Proc. 30th Int'l Conf. Information Technology Interfaces, Univ. Computing Centre, Zagreb, Croatia, pp. 235–246, 2008.

[22] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities," In Proc. of the 7th High Performance Computing and Simulation, Leipzig, Germany, 2009.

[23] Youseff, M. Butrico, and D. D. Silva, "Towards a Unified Ontology of Cloud Computing," Grid Computing Environments Workshop (GCE08), 2008.

[24] H.R. Motahari-Nezhad, B. Stephenson, and S. Singhal, "Outsourcing Business to Cloud Computing Services: Opportunities and Challenges," Technical Report HPL-2009-23, 2009.

[25] Pylot, http://www.pylot.org.

[26] Linux Virtual Server (LVS), http://www.linuxvirtualserver.org/.

[27] A10 networks, http://www.a10networks.com/.

[28] Citrix, http://www.citrix.com/.

[29] Round robin DNS (RRDNS),http://en.wikipedia.org/wiki/Round-robin_DNS.

[30] Greenberg, A., et al., "The Cost of a Cloud: Research Problems in Data Center Networks," CCR, v39, n1, 2009.

[31] L. Aversa and A. Bestavros,"Load balancing a cluster of Web servers using Distributed Packet Rewriting," In Proceedings of the 19th IEEE International Performance, Computing, and Communication Conference, 2000.

[32] X. Wang and W. Fu, "Research and implementation of the multiple exit load balancing of campus network based on DNS service", Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on , vol.3, no., pp.471-474, 10-12 June 2011.

[33] J. Mogul., "Network behavior of a busy Web server and its clients," Research Report 95/5, DEC Westem Research Laboratory, October 1995.

[34] D. M. Dias, W. Kish, R. Mukherjee, and R.Tewari, "A Scalable and Highly Available Web Server," Proc. 41st IEEE Computer Soc. Int'l Conf., IEEE Computer Soc. Press, Los Alamitos, Calif., Proceedings of IEEE COMPCON'96, pp.85–92, Feb 1996.

[35] M.Menzel and R. Ranjan,"CloudGenius: decision support for web server cloud migration," In Proceedings of the 21st international conference on World Wide Web (WWW '12), ACM, New York, NY, USA, pp. 979-988, 2012.