SIR (System Information Reporter) Framework for Performance Comparison of Different Operating Systems with KVM Hypervisor

Abhishek K.L [1], Anupama K.C [2]

[1] *M.Tech, IV Sem, Dept. of ISE, Bangalore Institute of Technology, Bangalore, Karnataka, India*

[2] *Assistant Professor, Dept. of ISE, Bangalore Institute of Technology, Bangalore, Karnataka, India*

---***---

Abstract-*Hypervisors enable multiple operating systems to run above it with the help of virtualization technique by sharing underlying physical resources. KVM (Kernel-based Virtual Machine) hypervisor is chosen here which uses hybrid virtualization technique i.e., it uses full virtualization technique along with hardware assisted virtualization. It is motivating to analyze different Operating Systems (OSs) performance with KVM hypervisor. We have chosen three guest operating systems, namely Windows 7, CentOS7 and Ubuntu 14.02 for the experimentation. The three Operating Systems (OSs) are prudently chosen to represent three categories namely Hardware virtualized gust, para-virtualized commercial guest and para-virtualized free guest. The Operating Systems are compared by performance tests of CPU utilization, memory management, disk activity, and network communication. Important System information is gathered using SIR (System Information Reporter) framework on the respective guest Operating System on KVM hypervisor.*

Keywords: virtualization, hypervisor, SIR, KVM.

I. INTRODUCTION

Virtualization technology is explanation of something a well known as became lost in basic material, computerized information devices, occupied systems or a server which will be created in virtual (rather than actual) way of doing thing for the usage. These technologies are accepted on a large scale in the society right to its expansion in show and tell the resources across the determined regions. Virtualization comes facing the romp where there is a World Wide Web connecting places to places. Virtualization allows active on many occupied systems at a presage which will give stage to associate those hired systems performance [1].

Hypervisors empower distributed computing model to give adaptable foundations and on interest access to processing assets as they backing numerous working frameworks to keep running on one physical server simultaneously. Hypervisor sellers do assert that they have invalidated virtualization overhead totally contrast with local framework, yet there exists certain moment virtualization overhead on the grounds that virtual machines need to speak with center layer hypervisor to get to the hidden physical equipment furthermore there is an effect of other virtual machines running on the same hypervisor Different virtualization systems like full virtualization, para-virtualization and cross breed model virtualization are utilized as a part of hypervisors advancement [2].

Performance evaluation involves four types of parameters, to be compared in three different stages.CPU performance assessment, network performance assessment, I/O read-write performance assessment, memory performance assessment are the four parameters which will be providing information about the operating systems performances. Three stages of comparison of performances are low workload, medium workload and high workload which will provide the performance results in three different levels leads to easy evaluation of performances. SIR framework will give the information about the CPU processing time, memory used and other valuable information as well, which can be plotted to find the results changes in the performance assessments of all the parameters which are involved [3].

II. RELATED WORK

P. VijayaVardhan Reddy, et-al [4]the paper, "Evaluation of different hypervisors performance in the private cloud" evaluates the performance of three hypervisors ESXi, XenServer's and KVM using SIGAR framework for system information and Passmark for system workloads in the private cloud environment. Private cloud has been designed using open source cloud computing software CloudStack. Hypervisors are deployed as hosts in the CloudStack. This paper recommends best suited hypervisors for respective workloads in the private cloud based on the performance of system information and system workloads.

From the test results KVM hypervisor scores better performance when compared to ESXi and XEN sever hypervisor.

Prakash P, et-al. [5] discussed on virtualization Machine Guest Operating System. There are focused on Hypervisor on type-2. Various Guest OS was keeping running on various- 2 Virtual Machines and checked their execution. Hypervisor is the center part of virtual machine (VM) framework and its adequacy incredibly impacts the execution of entire framework. This paper gives the execution of various visitor (virtual) working on same host working framework furthermore it talk about execution of two Virtualization hypervisors (sort 2) accessible for x86 engineering VMware Workstation and Virtual Box utilizing benchmark applications. ***From execution assessment the VMware Workstation has the best execution furthermore Windows 8 has the best execution as virtual working framework on both VMware and VirtualBox.***

Kim Thomas Moller, et al. [6]the paper "Virtual Machine Benchmarking" proposed another benchmark VMBench. He had utilized multi-stage approach while measuring the execution of a virtual machine environment. In the fundamental stage, VMBench puts down the signs and stores up the outcomes. Second stage utilizes a prompt model to foresee the come to fruition for constant applications. Third stage utilizes synchronous virtual machines under non-immaculate conditions to gage the execution. VMBench, rather than depending upon information throughput it depends on latency.

D. Huang, et al. [7] the paper "Performance Measuring and Comparing of Virtual Machine Monitors "had evaluated the performance of three hypervisors in the private cloud with single virtual machine. CPU utilization and Memory details is captured using SIGAR framework as absolute values. Disk activity is captured using Passmark [8] and network performance using Netperf [9]. In CPU utilization, ESXi scores low CPU utilization hence gives better performance compared to other two hypervisors. ***In available memory test, Xen Server's performance is noticeably better among three hypervisors***.

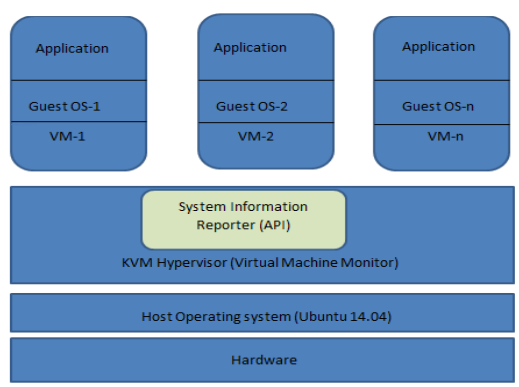
Kyu Ho Park, et al. [10]the paper "A Performance Comparison of Hypervisors" assesses the execution of two hypervisors ESXi and Xen Server's with comparative tests utilizing standard benchmarks. Hypervisors utilizing virtualization innovation empower numerous working frameworks (operating systems) to keep running on one physical server. The primary reason for these benchmarking tests utilizing SPEC CPU 2000, Passmark, and NetPerf was to utilizing SPEC cpu2000 [11], Passmark, NetPerf was to survey the execution of two hypervisors Xen and ESX. ***In view of the outcomes both hypervisors perform near local with the exception of ESX conveys somewhat better execution over Xen.***

L. Rajamani, et al. [12]the paper "Performance Evaluation of Operating Systems in the Private Cloud with Xen Server's Hypervisor Using SIGAR Framework" assesses the three working frameworks' (Windows 2008 R2 (Hardware Virtualized Guest or Hardware Virtual Machine), Red Hat Enterprise Linux (RHEL5 – para-virtualized business visitor) and Ubuntu 10.04 Lucid Lynx (para-virtualized free visitor) execution in the private cloud with Xen Server hypervisor utilizing SIGAR structure. ***From test outcomes for Xen Server's hypervisor, which utilizes para-virtualization method, both para-virtualized visitors (guests) Linux and Ubuntu displays marginally preferable execution over the equipment virtualized visitor Windows.***

After studying the appropriate work on operating systems (guest virtual machines) and hypervisors performance we have carefully chosen three different guest operating systems namely CentOS 7, windows 7 and Ubuntu 14.04 to evaluate their performances in the context of CPU utilization, Memory availability, Disk activity and Network communication.

III. SYSTEM ARCHITECTURE

The Figure 1 depicts the architecture of whole system which involves one server operating system is connected to guest operating systems with KVM hypervisor. The design shown is straight forward which connects all operating systems and performance will be gathered from single operating system at each time which will be done for all the three stages of workload. SIR framework will be implemented in all guest operating systems so that information will be gathered from there. At each workload given to an operating system which yields four different types performance



assessment. This should be repeated with remaining other two operating systems. For next stages of workload, work will be assigned to single operating system where other two operating systems will be utilizing system resources. Same has to be done with every operating system to get all the performance assessments.

Figure 1: Architecture of the proposed system.

The Figure 2 shows the connection of guest operating systems with the server operating systems contains KVM as a hypervisor. Three guest operating systems are, Windows 7, CentOS 7 and Ubuntu 14.04. Performance evaluation involves four types of parameters. CPU performance assessment, network performance assessment, I/O read-write performance assessment, memory performance assessment are the four parameters which will be providing information about the operating systems performances.

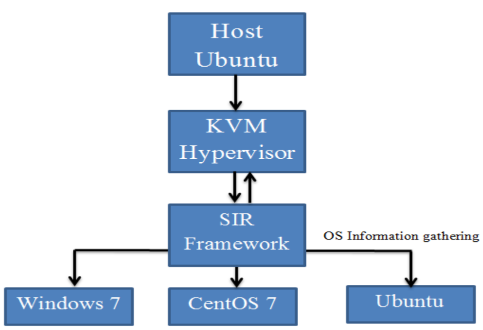


Figure 2: Connection of guest operating systems with server.

Figure 3 shows the three stages, which illustrate low, medium and high workloads [13] [14] respectively:

Stage 1: The visitor working framework (guest Operating System) execution is caught whenother two visitor working frameworks are in the idle condition (For example: Ubuntu is tested when CentOS and windows are idle).

Stage 2: The visitor working framework execution is caught when other one visitor workingframework consuming maximum System resources (For example: Ubuntu is tested when CentOS is idle and windows using maximum system resources).

Stage 3: The visitor working framework execution is caught when other two visitor workingframework consuming maximum System resources (For example: Ubuntu is tested when CentOS and Ubuntu using maximum system resources).

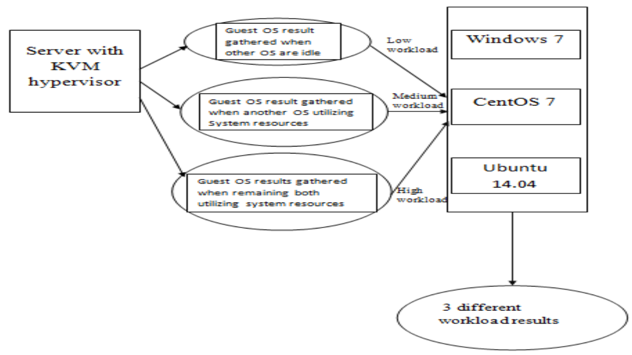


Figure 3: Different stages of workload.

As shown in Figure 4 the performance measure of three operating systems will be done on parameters like CPU utilizations, network assessment performance, memory assessment performance, and I/O read performance under three stages of workloads namely low workload, medium workload and high workload using SIR framework.

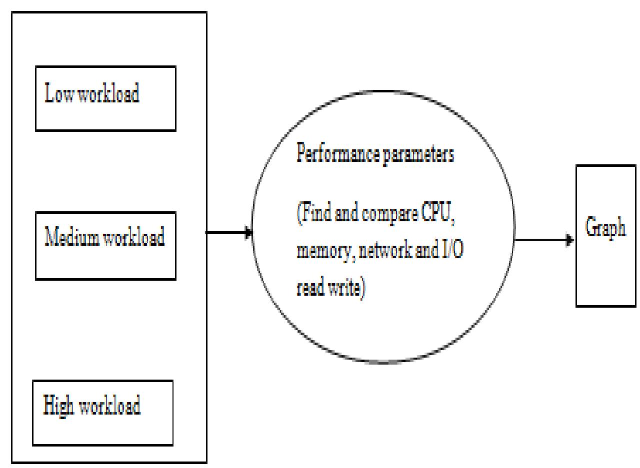


Figure 4: Workloads to be resulted in four parameters using graph.

IV. CPU PERFORMANCE ASSESSMENT

The CPU utilization assessment is carried out on the three guest operating systems namely windows 7, CentOS 7, Ubuntu 14.04 under three different workload which illustrates low, medium, high workloads. Producer-Consumer problem will run on all the three guest operating systems in order to evaluate the CPU utilization assessment under low, medium and high workloads respectively.

In order to get the processing time for CPU Utilization **getProcessCpuTime** method [15] has been used. **intprePerf=c.getProcessCpuTime();** has been used to get the processing time before the execution of the process. Processing time for before execution of process has been captured because if any unknowing services are using the CPU. **intpostPerf=c.getProcessCpuTime();**has been used to capture the processing time after the execution of the process.

V. MEMORY PERFORMANCE ASSESMENT

Quick sort program for random numbers (greater than 100000 elements) will run on all the three guest operating systems in order to evaluate the Memory performance assessment under low, medium and high workloads respectively. The following code snippet is used to get the Free Physical Memory.

In order to get the Memory performance assessment **getFreePhysicalMemorySize();** method [16] has been used. **longprePerf = c.getFreePhysicalMemorySize();**has been used to get the Free Physical memory (in Bytes) before the execution of the process. Free Physical memory for before execution of process has been captured because if any unknowing services are using physical memory. **IntlongpostPerf = c.getFreePhysicalMemorySize ();** has been used to capture the Free Physical memory (in Bytes) after the execution of the process.

VI. DISK I/O READ-WRITE PERFORMANCE ASSESSMENT

The Disk I/O Rea/Write performance assessment has been carried out at three different stages namely Low workload, Medium workload and high workload on all the three guest operating systems by running the program which reads and writes the Video file. The following code snippet is used to get the processing time for Disk I/O read/write.

In order to get the processing time for Disk I/O Read-Write **getProcessCpuTime** method [17] has been used. **intprePerf=c.getProcessCpuTime();** has been used to get the processing time before the execution of the process. **intpostPerf=c.getProcessCpuTime();**has been used to capture the processing time after the execution of the process.

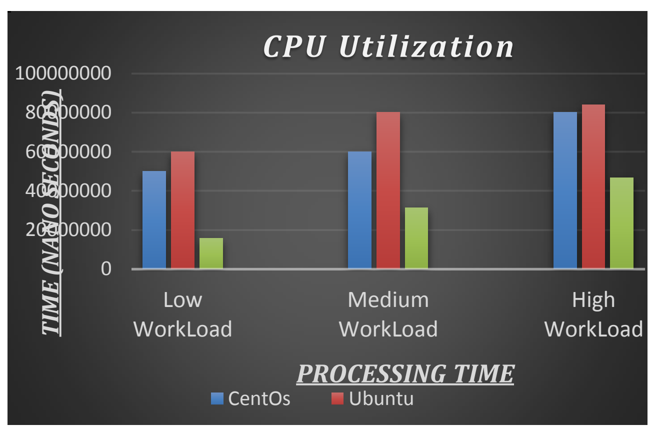
VII. RESULTS

This segment provides detailed results of all the performance tests which are executed on three guest operating systems using SIR API. In CPU utilization lower CPU consumptions (less processing time) is better for guest OS. In the case of memory tests, lesser memory used (in bytes) indicates superior performance of guest OS. In case of Disk I/O read-write, lesser processing time (in Nano seconds) are the signs of better guest OS on KVM hypervisor. In case of network performance lower latency (in Nano seconds) indicates better performance for guest OS.

A. CPU UTILIZATION RESULTS

The CPU utilization assessment is carried out on the three guest operating systems namely windows 7, CentOS 7, Ubuntu 14.04 under three different workload which illustrates low, medium, high workloads. Producer-Consumer problem will run on all the three guest operating systems in order to evaluate the CPU utilization assessment under low, medium and high workloads respectively. In order to get the values for CPU utilization assessment the difference between the two processes that is before process (before the execution of particular process) and after process (after execution of the particular process) has been taken as time taken for process (in Nano seconds),

Figure 5shows that Windows operating system performs very well under all the three stages of workloads when compared to Ubuntu and CentOS operating systems. The analyzing is done according to the less time taken by



respective operating systems to complete the process (in Nano seconds).

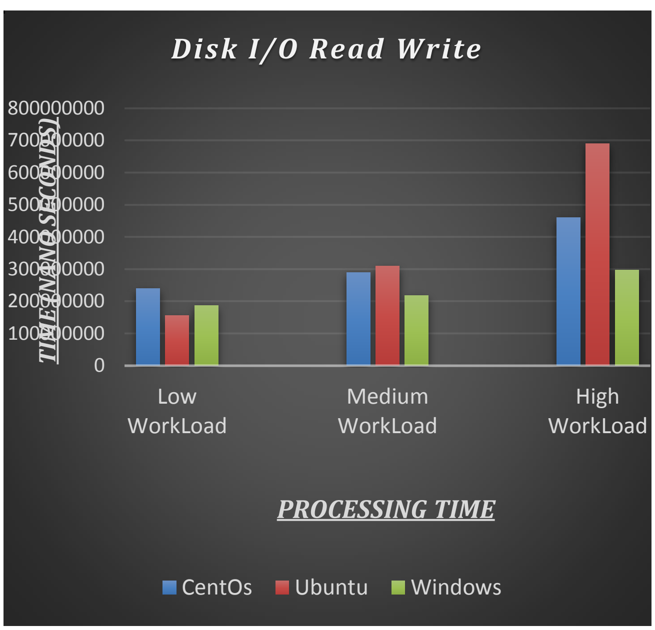
Figure 5: CPU Performance of guest operating systems (Lower value is better).

B. DISK I/O READ-WRITE PERFORMANCE RESULTS

The Figure 6 shows the disk I/O Read-Write performance assessment of three guest operating systems under three stages of workload by running the program which reads and writes the Video file on the test guest operating system Results on the respective guest operating systems are captured using SIR API. Using these values the comparison of the three operating systems has been evaluated using the graph as shown in Figure 6.

Disk I/O virtualization involves managing the routing of I/O solicit between virtual devices and shared mundane hardware. Windows is hardware virtualized guest operating system which has different kernel involved functionalities compared to the Linux based kernels. Some of the functionalities such as “CeVirtualSharedAlloc” which is built inside the kernel whereas it is built outside the kernel in Linux based systems.

Hence the I/O virtualization overhead is more in Linux based operating systems. Key to effective I/O virtualization is to keep the CPU utilization to minimum. Since Windows gives better performance for CPU utilization and hence Windows gives better Disk I/O Read-Write performance guest operating systems Ubuntu and CentOS Under all the three stages of workload.



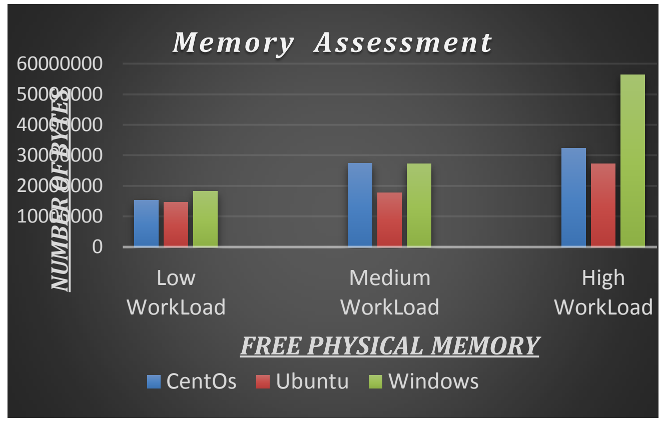


Figure 6: Disk I/O Read-Write Performance of guest operating systems (Lower value is better).

C. MEMORY PERFORMANCE RESULTS

Memory virtualization involves show and tell under the sun system recollection and dynamically allocating it to virtual machines. Virtual machine hallucination virtualization is literally similar to the virtual hallucination corroborate provided by new operating systems. Applications manage a contiguous devote space especially not originally tied to the concealed under the sun memory in the system. The disk keeps mappings of virtual page numbers to physical page numbers concentrated in page tables. The Figure 7 shows the memory performance assessment of three guest operating systems under low workload. In order to get the value for Memory assessment, the quick sort for random numbers generation (greater than 100000 elements) is run on all the guest operating systems. Hence from Figure 7 it is clear that Ubuntu 14.04 does well for memory usage on all the three stages of workload when compared to Ubuntu 14.04 and Windows 7.

Figure 7: Memory Performance of guest operating systems (Lower value is better).

D. NETWORK PERFORMANCE RESULTS

The Figure 8 shows the Network performance assessment of three guest operating systems result is captured by sending/receiving the mail with text or image or video attachment on the test guest operating system. In order to get the values for Network assessment the starting time of the process has been taken as zero. The time taken for completion of process is captured as latency. From the test results it shows that CentOS gives better network performance when compared to other two operating systems under low workload. As the workload increases i.e. under medium and high workloads, Ubuntu gives the better network performance.

The network drivers in windows and Linux based kernels are differently allocated with respect to the kernel. The network drivers in Linux based kernel are built inside the kernel whereas network drivers in windows are built outside the kernel which makes the network performance slow in windows as the network process has to communicate with the kernel from outside. Hence Ubuntu gives better network performance.

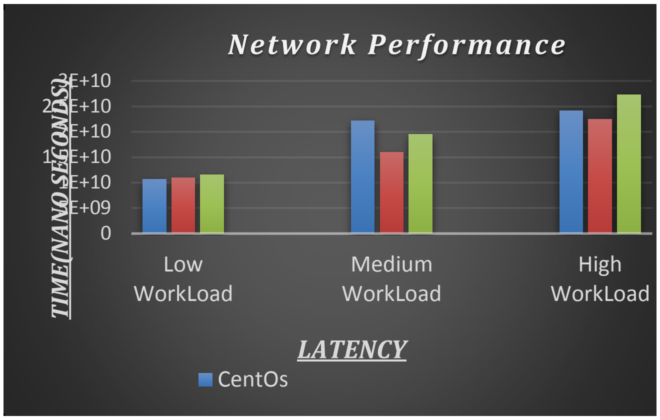


Figure 8: Network Performance of guest operating systems (Lower value is better).

VII. CONCLUSION

The intent of the paper is to evaluate and compare the performances of three guest operating systems to find out the best operating system by using KVM hypervisor and SIR framework. SIR is an application programming interface for accessing the operating system and hardware level information in different languages. In the experiment, java program has been written to gather system information using sir API by deploying libsigar-amd64-linux.so and sir-amd64-winnt.dll for linux and windows respectively. Three different operating systems have been chosen for evaluation, which are windows 7(hardware assisted guest), centos 7 and Ubuntu 14.04 (paravirtualized guests) as guest operating systems.

Test setup was challenging and collecting information through SIR API was a new idea. From the test results with KVM as a hypervisor, which uses hybrid virtualization technique, it is evident that windows 7 is better for CPU utilization and disk I/O read-write and Ubuntu is better for memory and network performance. overall three operating systems perform close to each other in all tests, except paravirtualized guest Ubuntu scoring marginally better over other two operating systems because it is less variant and more consistent. As a part of the future enhancement, more parameters can be added with the current parameters, micro benchmarks can be used to get precise results, and more operating systems can be evaluated from performances perspective.

REFERENCES

[1]. CloudStack – OpenSource Cloud Computing platform from Apache organization [Online] http://cloudstack.apache.org

[2]. Nanda, S., T. Chiueh, ―A Survey on Virtualization Technologies, Technical report, Department of Computer Science, SUNY at Stony Brook, New York, 11794-4400, 2005.

[3]. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I. (2009) "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility". In: Future Generation Computer Systems, Elsevier B.V.

[4]. P.VijayaVardan Reddy "Evaluation of Different Hypervisors Performance in the Private Cloud with SIGAR Framework", International Journal of Advanced Computer Science and Applications, Vol.5, No.2, 2014.

[5]. Prakash P and Biju R Mohan., "Evaluating Performance of Virtual Machine on Hypervisor (Type -2)," Association of Computer Electronics and Electrical Engineers, 2013 Proc. of Int. Conf. on Emerging Trends in Engineering and Technology, pp.1,9, 15-17 Jan. 2014.

[6]. Kim Thomas Moller, K.T., ―Virtual Machine Benchmarking, Diploma Thesis, Karlsruhe Institute of Technology, 2007.

[7]. D. Huang and J. Che, Q. He, Q. Gao "Performance Measuring and Comparing of Virtual Machine Monitors", College of Computer Science, Zhejiang University, Hangzhou 310027, China, IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, **(2008)**.

[8]. http://www.passmark.com/products/pt.html

[9]. Netperf [Online] http://www.netperf.org/netperf/

[10]. Kyu Ho Park &Qinming 'Performance Evaluation of Hypervisors in the Private Cloud based on System Information using SIGAR Framework and for System Workloads using Passmark', International Journal of Advanced Computer Science and applications Technology, Vol 70 (2014), pp.17-32. Standard Performance Evaluation Corporation (SPEC). [Online] http://www.spec.org/

[11]. Standard Performance Evaluation Corporation (SPEC). [Online] http://www.spec.org/

[12]. L Rajamani, P.V.V.Reddy, "Performance Evaluation Of operating systems in the private cloud with Xen sever hypervisor using SIGAR framework", 9th IEEE International Conference on Computer Science and education (IEEE ICCSE 2015) August 22-24, 2015.

[13]. FUJITSU, Benchmark Overview-vServCon, white paper, March 2010.

[14]. XenSource (2007) A Performance Comparison of Commercial Hypervisors. XenEnterprise vs. ESX Benchmark Results. 2007 XenSource.

[15]. Apparao, P. &Makineni, S. & Newell, D.Virtualization (2006) Characterization of network processing overheads in Xen. Technologyin Distributed Computing, 2006. VTDC 2006.

[16]. Jianhua, C. &Qinming, H. &Qinghua, G. &Dawei, H. (2008) Performance Measuring and Comparing of Virtual Machine Monitors. Embedded and Ubiquitous Computing, 2008. EUC '08.

[17]. Menon, A. et Al. (2005) Diagnosing Performance Overheads in the Xen Virtual Machine Environment. Conference on Virtual Execution Environments (VEE'05).

[18]. Shan, Z. &Qinfen, H. (2009) Network I/O Path Analysis in the Kernel- based Virtual Machine Environment through Tracing. Information Science and Engineering (ICISE).

[19]. Jianhua, C. &Qinming, H. &Qinghua, G. &Dawei, H. (2008) Performance Measuring and Comparing of Virtual Machine Monitors. Embedded and Ubiquitous Computing, 2008. EUC '08.

[20]. P.V.V.Reddy, Lakshmi Rajamani, “EvaluationOf different operating systems performance in the private cloud with ESXi hypervisor using SIGAR framework”, 5th IEEE International Conference CONFLUENCE The next generation Information Technology summit (IEEE CONFLUENCE 2014) September 25-26, 2014.