Recommendation of Hypervisors for System Resources Intensive Application Workloads based on their Performance in the Private Cloud

P. VIJAYA VARDHAN REDDY, and LAKSHMI RAJAMANI
Osmania University INDIA

Hypervisors enable cloud computing model to provide scalable infrastructures and on-demand access to computing resources as they support multiple operating systems to run on one physical server concurrently. This mechanism enhances utilization of physical server thus reduces server count in the datacenter hence driving the benefits of reduced IT infrastructure setup and maintenance costs along with power savings. The paper evaluates the system resources (CPU, Memory, Network, and Disk I/O) performance of three hypervisors, namely ESXi, XenServer and KVM in a CloudStack environment using the SIGAR-Framework for consolidated application workloads with concurrently running virtual machines. Based on the results, the paper recommends the hypervisors suitability for system resources intensive applications. The three hypervisors are carefully chosen to represent three categories (full virtualized, para-virtualized, and hybrid virtualized). We have created a private cloud using open source cloud computing software CloudStack. Hypervisors are deployed as hosts in the CloudStack in the respective clusters. Consolidated workloads are generated using open source load generator tools on multiple virtual machines of hypervisors and important system resources information is gathered using SIGAR framework.

Keywords: CloudStack, Full virtualization, Hypervisor, Hybrid model virtualization, Para-virtualization, Private cloud, SIAGR, Virtualization

1. INTRODUCTION

Cloud computing as a model enables dynamic access to servers, networks, applications, services, storage and provides the option to pay only what has been used for [Mell et al. 2009]. The major benefits of cloud computing are flexible and scalable infrastructures, reduced implementation and maintenance costs, IT department transformation (focus on innovation than maintenance and implementation) and increased availability of high performance applications to small/medium sized businesses.

Cloud computing model promotes availability and is composed of four deployment models. Public cloud model usually deployed over the internet and made available to all. In the community cloud model, the cloud infrastructure shared for specific community. Private clouds are deployed behind the firewall of a company and the cloud infrastructure is operated solely for an organization. Composition of two or more cloud models forms a hybrid cloud model. Private cloud deployment model creates the proprietary computing architecture behind a firewall with full control over infrastructure. We have created a private cloud for the experiment.

Virtualization Technology plays an important role in the success of cloud computing. The technology enables optimization of complex IT resources in a scalable way thus the delivery of services is simplified. Virtualization is a technology that combines or divides computing resources to present many operating environments using methodologies like hardware and software partitioning, machine simulation, emulation, time-sharing, and many others [Nanda et al. 2005].

Cloud computing allows customers to reduce the cost of the hardware by allowing resources on demand. The SLA (Service Level Agreement) between the providers of cloud and the customers ensures that the service will be delivered properly [Buyya et al. 2009].

Hypervisor as a virtualization layer provides an infrastructural support to multiple virtual machines above it by virtualizing hardware resources such as CPU, Memory, Disk, and NIC. Hypervisors can be categorized into three models full virtualized hypervisor, para-virtualized hypervisor and hybrid model hypervisor based on the virtualization techniques that are used in their development. Citrix XenServer is a server virtualization platform built on the Xen Hypervisor. Xen [Xen 2009] uses para-virtualization technique. Para-virtualization modifies the guest operating system. XenServer provides a virtual infrastructure solution [Fujitsu 2007]. VMware ESXi Hypervisor uses full virtualization [Hostway, UK 2011] technique. The hypervisor installs all the hardware drivers and related software into the guest operating system. It traps every instruction that attempts to update hardware data structures [VMware 2007]. Hence, an extra level of mapping is in the page table [Barham et al. 2013]. KVM is a hybrid model hypervisor. KVM provides virtualization capability to guest process with user and kernel modes [Shan et al. 2009]. It uses all Linux capabilities in memory and I/O scheduling without reinventing the wheel. KVM uses hardware-assisted virtualization capabilities along with full virtualization technique hence depicts a hybrid model.

This paper evaluates the performance of three hypervisors VMware ESXi 4.1, Citrix Systems XenServer 6.0, and KVM (Ubuntu 12.04 Server) for consolidated application workloads in the private cloud environment using SIGAR (System Information Gatherer and Reporter) framework. We have created the private cloud using open source cloud computing software CloudStack. The paper evaluates the hypervisors system resources performance for the consolidated application workloads with concurrently running virtual machines (VMs). The important system resources information is gathered using SIGAR framework and the paper recommends the hypervisors suitability for system resources intensive application workloads. This paper evaluates the performances of three hypervisors which are carefully chosen to represent full, para, and hybrid virtualization techniques. This paper is novel because it conducts performance tests of hypervisors for CPU utilization, Memory management, Disk activity and network performance with consolidated application workloads generated using open source load generator tools like Httperf, JMeter, and Mysqlslap. System information is gathered using SIGAR framework on all three hypervisors.

The results and recommendations in this paper will help both IT decision makers and end users to choose the right virtualized hypervisor for the respective system resources intensive application workloads in their private cloud environments. Findings in the paper will also help the hypervisor vendors to notice and mitigate the respective system resources performance shortcomings in comparison with other hypervisors.

2. RELATED WORK

The following papers are cited to study the relevant work which had happened in the selected research area.

A Performance Comparison of Hypervisors [VMware white paper 2007] paper by VMware conducts different performance tests to measure the performance and scalability of two hypervisors ESX and Xen. A Performance Comparison of Commercial Hypervisors [XenSource 2007] paper by XenSource also conducts same performance tests to evaluate the performance of both hypervisors ESX and Xen. In the experiments, to evaluate CPU performance of two hypervisors for CPU intensive applications, they have used Standard Performance Evaluation Corporations (SPEC) SPECcpu2000. To evaluate CPU and Memory performance of hypervisors for typical system workloads they have used Passmark benchmark. They have used SPECjbb2005 to evaluate hypervisors performance for an application servers workload and SPECcpu2000 INT to assess the two hypervisors performance for development workloads. They have also used Netperf to evaluate network performance of both hypervisors. In the results, they compared both hypervisors with native and claimed that both hypervisors give near native performance except ESX scoring slightly better performance over Xen.

A Performance Comparison of Hypervisors for Cloud Computing [Suganya et al. 2012] paper by Suganya, evaluates the performance of VMware ESXi 4.1, Citrix Systems Xen Server 5.6 and Ubuntu 11.04 server KVM hypervisors using SPECvirt_sc2010v1.01. Results indicate that ESXi scores better performance over other two hypervisors.

The above two papers have used standard benchmarks, namely SPEC and Passmark.

Benchmark Overview vServCon a white paper by Fujitsu PRIMERGY Servers [FUJITSU 2010] talks about vServCon benchmark which was developed for their internal purpose to measure and assess performance of virtualized servers. According to them, vServCon is not a new benchmark, but a framework that consolidates already established benchmarks, as workloads, in order to simulate the load of a virtualized consolidated server environment. Three applications database, application server, web server executed on each virtual machine and all these applications are stressed with load generators through established benchmarks. All individual results are summarized in one result and they named it as a score for the performance capability of a virtualized environment.

The above paper evaluates the hypervisors performance for consolidated workloads using established benchmarks like SPECjbb2005, Sysbench, and WebBench.

Different hypervisors such as XEN, KVM and VMware ESX performances have been evaluated to measure the overhead of virtualization with different toolkits. Menon had used a toolkit called Xenoprof (system-wide statistical profiling toolkit for Xen virtual machine environment similar lines of OProfile for Linux) to evaluate the performance overhead of network I/O devices. Menon [Menon et al. 2005] had used Xenoprof to debug Xen and been able to improve the network performance. Menon claims from his research that domain0 performance is close to native, but guest operating system performance degrades considerably because of high CPU utilization. Jianhua [Jianhua et al. 2008] had used LINPACK benchmark tool to test processing efficiency on floating point. Jianhua had observed that windows XP gives better performance than fedora 8 on Xen. Jianhua had clarified it as Xen owns certain enhancement packages for windows XP than fedora. Jianhua had used LMbech to evaluate memory virtualization of Xen and KVM and noticed that Xens performance is better than that of KVM. Jianhua had used IOzone to compare file system performance among Xen and KVM. Jianhua had found that without intel-VT processor the performance of Xen and KVM is significantly slower than that of native. With intel-VT processor, Xen performance is significantly improved but not of the KVM because KVM does not utilize the functionalities of the intel-VT processor.

The above papers have used standard toolkits to evaluate the performances of hypervisors.

Virtual Machine Benchmarking, by Kim Thomas Moller [Moller et al. 2007] talks about the creation of a new benchmarking suite VMbench and how it evaluates the performance of virtual machine environments. VMbench uses three stages of analysis. After defining hypervisor performance signature with micro and nano-benchmarks, a virtual machine exercises defined operations so that the performance of virtualization-specific functional primitives are accurately measured. Second stage uses best-case predictions for realistic applications using a linear model. After introducing virtual machine interference with concurrent VMs, the performance is measured for non-optimal conditions. VMbench follows a latency-oriented approach.

The above paper used new benchmarking suite for latency-oriented approach and ignores data specific approach.

Performance Comparison of Hypervisors in the Private Cloud, by P.V.V.Reddy [Reddy V V et al. 2014] conducted experiments in the private cloud with CloudStack for three hypervisors for system workloads using Passmark benchmark. This paper evaluated three hypervisors performance for CPU utilization, Memory management, Disk activity using Passmark and Network communication performance using Netperf. From the results when single virtual machine (VM) is running for system workloads, ESXi hypervisors performance in CPU utilization is better than that of XenServer and KVM. In Memory and Disk activity performance XenServer scores marginally better over other two hypervisors. In Network communication both XenServer and ESXi scores equal to native without any virtualization overhead.

The above paper evaluates the hypervisors performance with single virtual machine for system workloads using standard benchmarks like Passmark and Netperf.

Evaluation of Different Hypervisors Performance in the Private Cloud with SIGAR Framework by P.V.V.Reddy [Reddy V V et al. 2014] evaluates the performance of three hypervisors in the private cloud with single virtual machine running on three hypervisor hosts. CPU utilization, Memory details are captured using SIGAR framework with absolute values. Disk activity is captured using Passmark and network communication using Netperf. In CPU utilization, ESXi scores low CPU utilization hence gives better performance compared to other two hypervisors. In available memory test, XenServer performance is noticeably better among three hypervisors.

The above paper evaluates the hypervisors performance with single virtual machine for system workloads using SIGAR benchmark for CPU, memory utilization and Passmark benchmark for disk activity and Netperf benchmark for network communication.

Performance Evaluation of Hypervisors in the Private Cloud based on System Information using SIGAR Framework and for System Workloads using Passmark by P.V.V.Reddy [Reddy V V et al. 2014] evaluates virtualization overhead of three hypervisors compared to the native system with a single virtual machine.

The above paper compares the hypervisors performance (single virtual machine for system workloads) with native machine using SIGAR and Passmark to point out virtualization overhead.

The observed drawbacks from literature survey are most of the papers have evaluated hypervisors performance as a single entities by ignoring the fact that these hypervisors are eventually going to be used in the cloud environment. Most papers have used standard benchmarks and tried with single virtual machine to snub the realities that now in the industry all hypervisors run multiple virtual machines. To overcome these drawbacks we have created a private cloud for the experiment and we relied on multiple virtual machines to generate consolidated application workloads and used most sought open source tools for load generation and data collection.

In the current work we have chosen open source load generator tools (Httperf, JMeter, Mysqlslap) to generate consolidated applications workloads and we have selected SIGAR framework to gather system resources information from concurrent virtual machines of a hypervisor. Paper [Moller et al. 2007] relies on latency oriented and we prefer to choose data throughput which makes this paper pragmatic. Paper [Reddy V V et al. 2014] evaluates performance of hypervisors in the private cloud with Passmark benchmark for system workloads with single virtual machine. Paper [Reddy V V et al. 2014] evaluates performance of hypervisors using SIGAR with absolute values using single virtual machine. Paper [Reddy V V et al. 2014] evaluates performance of hypervisors using SIGAR for virtualization overhead compared to native using single virtual machine. The current experiment is distinctive because it conducts an experiment with concurrently running virtual machines with applications like Web Server, Application Server, and DB Server. System resources information is gathered on all three application VMs for respective hypervisor. Based on the results we have recommended the suitability of the hypervisors for the respective system resources intensive application workloads which will be useful to industry and academia.

After analyzing the relevant work on hypervisors performance, we have chosen SIGAR framework to evaluate the system resources performance of three hypervisors for concurrently running Virtual Machines. Workloads are generated simultaneously on three applications web server, application server, and the database server which are running as three concurrent VMs. These three virtual machines are deployed on each hypervisor and all applications are stressed with open source load generator tools and system information is gathered from three hypervisors using SIGAR framework. Private Cloud setup is simulated because it is more secured. Private Cloud is deployed behind the firewall of an organization and the infrastructure will be operated exclusively for the specified organization, hence it has fewer security concerns, unlike the public cloud. SIGAR API has used because it gives minute system information very accurately.

3. EXPERIMENTAL SETUP - PRIVATE CLOUD: CLOUDSTACK WITH HYPERVISORS

The experimental set up is simulated by creating the private cloud using CloudStack. CloudStack is an Infrastructure as a service (IaaS) cloud-based software which builds private cloud environments. CloudStack supports multiple hypervisors like KVM, XenServer, Hyper-V, Oracle VM, Bare Metal and VMware ESXi. CloudStack has the ability to build cloud environments with different hypervisors with web interface for users and administrators. CloudStack is open source software written in java that is designed to deploy and manage large networks of virtual machines, as a highly available, scalable cloud computing platform. CloudStack offers a web interface, command line to manage the cloud environment.

In the private cloud, the Management Server runs on a dedicated server or a VM. It controls allocation of virtual machines to hosts and assigns storage and IP addresses to the virtual machine instances. The Management Server runs in a Tomcat container and requires a MySQL database for persistence. In the experiment, Management Server (a Virtual Machine with hardware configuration of 4GB RAM and 100GB hard disk) is installed on Ubuntu (12.04 64-bit) operating system.

Host machines, where hypervisors are installed on a bare metal with hardware configuration of AMD FX 8150 8 core 3.6 GHz processor, 32 GB RAM, 1 TB hard disk and 2 NICs for the test environment. Hardware configuration of host machines is depicted in Table I.

Table I: HARDWARE CONFIGURATION OF HYPERVISORS

Hardware Profile	ESXi	XenServer	KVM
CPU	AMD FX 8150 8 core, 3.6 GHz	AMD FX 8150 8 core, 3.6 GHz	AMD FX 8150 8 core, 3.6 GHz
RAM	32 GB	32 GB	32 GB
Hard Disk	1 TB	1 TB	1 TB
NICs	2	2	2

Front end will be any base machine to launch CloudStack UI using web interface (with any browser software IE, Google Chrome) to provision the cloud infrastructure by creating zone, pod, cluster and host in the sequential order.

In our test environment XenServer 6.0, ESXi 4.1 and KVM (Ubuntu 12.04) hypervisors are deployed as hosts and three virtual machines (VM1, VM2, VM3) with respective applications like VM1-Webserver, VM2-Application Server and VM3-Database Server are installed on all three hypervisors in the private cloud as depicted in figure 1.

Three applications are installed on three virtual machines on each hypervisor and load is generated by the respective applications using open source load generators. Httperf is an open source load generator tool for the web server. JMeter is a GUI based tool which allows users to generate a heavy load on the individual servers to the group of servers. Mysqlslap is a problem-solving program designed to imitate client load for a MySQL server. After generating consolidated loads with load generator tools the system resources information of hypervisors in the context of CPU utilization, Memory management, Disk activity and Network communication is captured on all the three hypervisors using SIGAR API.

3.1 Hardware configuration and Load profile for respective applications

Each Hypervisor host machine contains the hardware configuration of AMD FX 8150 8 Core 3.6 GHz processor, 32 GB RAM, 1 TB hard disk. Three VMs are allocated 2vCPUs each, 8 GB RAM and 250 GB hard disk as depicted in the Table II.

Respective operating systems and application are installed in three virtual machines as mentioned in Table II. Httperf is used to generate load on the web server with load profile of 10,000 sessions, 100 sessions per second and with 4 calls per session. JMeter is used to generate load on the application server with 100 concurrent hits to business logic bean. Mysqlslap is used to create load on the database server with 10,000 queries and 100 concurrent clients with final load profile of 100 queries per client.

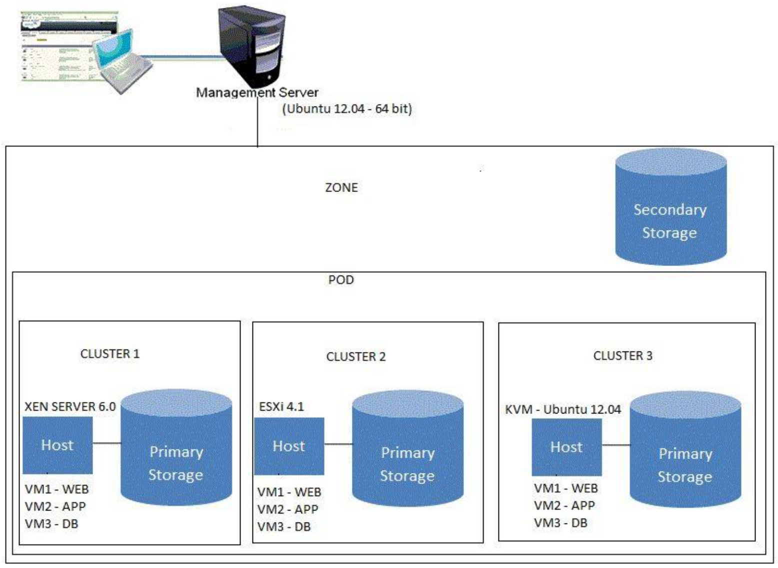


Figure. 1: Experimental Setup Private Cloud (CloudStack with Multiple hypervisors).

Table II: HARDWARE CONFIGURATION AND LOAD PROFILE OF DIFFERENT VIRTUAL MACHINES ON EACH HYPERVISOR

Hardware and Software Profile	VM1	VM2	VM3
Application	Web Server	Application Server	Database Server
Software	Apache 2.4	GlassFish 4.0	MySQL 5.7
Operating System	RedHat Enterprise Linux RHEL 5 - 64bit	RedHat Enterprise Linux RHEL 5 - 64bit	Windows 2008 R2 Server - 64 bit
vCPU	2	2	2
Memory	8 GB	8 GB	8 GB
Disk Capacity	250 GB	250 GB	250 GB
Load Tool	Httperf	JMeter	Mysqlslap
Load	sessions = 10,000 and rate = 100 sessions/second and 4 calls / session	100 concurrent invocations	concurrency = 100 and number of queries = 10,000 and queries per client = 100
Performance Measured using	SIGAR	SIGAR	SIGAR

After generating the load on each application on respective VMs of three hypervisors, the CPU, Memory, Disk I/O, and Network performances are captured using SIGAR framework. SIGAR is a platform independent tool for accessing system level information in Java and other programming languages. In the experiment, Java program has written to gather system information using SIGAR API by deploying sigar-amd64-winnt.dll for Windows and libsigar-amd64-linux.so for Linux.

4. RESULTS

This section provides the detailed results of all the performance tests which are executed on three hypervisors using SIGAR API. In CPU test for consolidated workloads, low utilization of CPU is better for a hypervisor. In the case of Memory test for consolidated workloads, low utilization of memory indicates superior performance of a hypervisor. In the case of Disk I/O tests for applied workloads higher read and write are the signs of a quality hypervisor. In Network performance for client-send and receive tests, higher transfer rate with given workloads indicates the better performance of a hypervisor.

4.1 CPU Performance

CPU utilization on the virtual machine for the respective application workload is captured when it is running on the particular Hypervisor. CPU utilization details are captured through java program using SIGAR API on the virtual machine operating system for each hypervisor. CPU utilization is gathered for all three virtual machines on the respective hypervisor for generated workloads. Low utilization of CPU indicates the better performance of a hypervisor.

CPU utilization is gathered from the following code snippet in the java program using SIGAR API. CpuPerc perc = sigar.getCpuPerc(); [Object sigar is an instance of SIGAR class]

From figure 2, ESXi shows 8.8% CPU utilization, XenServer shows 9.6% CPU utilization and KVM shows 12.6% CPU utilization. From CPU system resources performance perspective, ESXi delivers better performance for the consolidated application workloads compared to other two hypervisors.

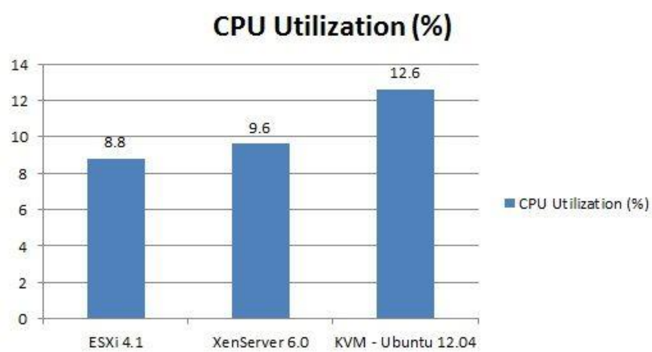


Figure. 2: CPU utilization captured using SIGAR (Lower Value is better).

ESXi scores low CPU utilization hence gives better CPU performance compared to other two hypervisors. XenServers CPU utilization is more than ESXi and less than that of KVM. Hence, XenServer gives better CPU performance compared to KVM but less than ESXi.

Due to virtualization overhead hypervisors need to handle more CPU instructions compared to the native machine. Among three hypervisors, ESXis virtualization technique is superior hence it gives low CPU utilization compared to other two hypervisors. Because of XenServers virtualization technique, it has to handle more CPU instructions hence it has impacted its CPU performance compared to ESXi. KVM falls behind two hypervisors with its virtualization technique and needs improvement from CPU performance perspective. Hence for CPU intensive applications, ESXi is a best-suited hypervisor.

4.2 Memory Performance

Memory utilization on the virtual machine for the respective application workload is captured when it is running on the particular Hypervisor. Memory utilization details are captured through java program using SIGAR API on the virtual machine operating system for each hypervisor. Low utilization of memory indicates the better performance of a hypervisor.

Used memory, free memory as long objects are gathered from the following code snippet in the java program using SIAGR API. long used = sigar.getMem().getUsed(); long free = sigar.getMem().getFree(); [Object sigar is an instance of SIGAR class]

From figure 3, ESXi shows 47% memory utilization, XenServer shows 45.9% memory utilization and KVM shows 51.3% memory utilization. From memory performance perspective, XenServer delivers better performance for the consolidated application workloads compared to other two hypervisors.

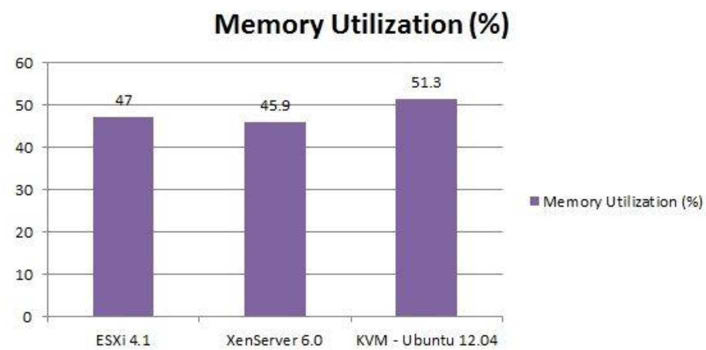


Figure. 3: Memory utilization captured using SIGAR (Lower Value is better).

XenServer scores low memory utilization hence gives better memory performance compared to other two hypervisors. ESXis memory utilization is more than XenServer and less than that of KVM.

Because of ESXis virtualization technique, it has an overhead of trapping, scanning and translation which eventually impacted its memory performance compared to XenServer. Overall KVM falls behind two hypervisors and it needs to improve on memory performance perspective. Hence for memory intensive applications, XenServer is a best-suited hypervisor.

4.3 Disk I/O Performance

The following figure 4 shows results gathered for the Disk I/O read-write tests. The size of the test file we used is 16 GB which is half of the servers RAM (32 GB). Read Write performance results are captured using Java program with SIGAR API on the separate windows 2008 R2 guest operating system while the same workloads are generated on three virtual machines on each hypervisor. Results indicate that XenServer performance is slightly better than that of ESXi. KVM falls significantly lower than other two hypervisors. High disk I/O rate indicates the better performance of a hypervisor.

readBytes, writeBytes as long objects are gathered from the following code snippet in the java program using SIAGR API. long readBytes = sigar.getDiskUsage(driveName).getReadBytes(); long writeBytes = sigar.getDiskUsage(driveName).getWriteBytes(); [Object sigar is an instance of SIGAR class]

In Disk I/O Read-Write performance experiment, in Read test ESXi records 121 Mb/s speed, XenServer records 132 Mb/s speed and KVM records as 100 Mb/s speed with load generation on the three virtual machines. In Write test ESXi records 112 Mb/s speed, XenServer records 123 Mb/s speed and KVM records as 91 Mb/s speed with load tests are on. Under normal conditions hypervisors records disk I/O speed of 240 Mb/s read and 235 Mb/s write for given hardware configuration but with generated loads on three virtual machines hypervisors disk I/O performance has come down 50% of their original capacity. As the load is same on all three hypervisors and for generated stress, XenServer exhibits better disk I/O performance compared to other two hypervisors.

XenServer scores high read-write rate hence gives better disk I/O performance compared to other two hypervisors. ESXis read-write performance is less than XenServer and more than that of KVM. Because of ESXis virtualization technique, it has an overhead of trapping, scanning

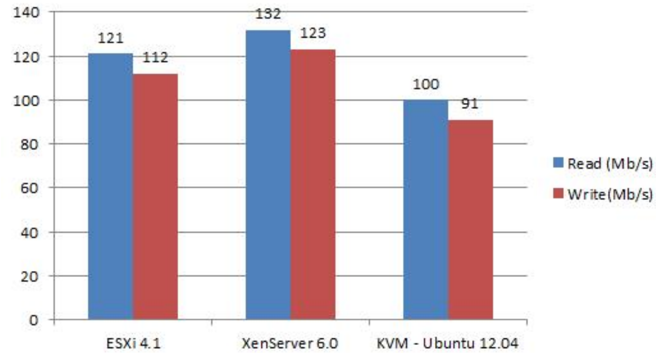


Figure. 4: Disk I/O Read-Write performance captured using SIGAR (Higher values are better).

and translation which eventually impacted its read-write performance compared to XenServer. Overall KVM falls behind two hypervisors with its virtual I/O drivers slow performance. Hence for disk I/O intensive applications, XenServer is a best-suited hypervisor.

4.4 Network Performance

The following figure 5 shows results gathered for Network performance test for client-send and receive tests. Client-send and client-receive tests are conducted on the separate windows 2008 R2 guest operating system while same workloads are generated on three virtual machines on each hypervisor. Send, Receive transfer rate results are captured using Java program with SIGAR API on the separate guest operating system. Results indicate that ESXi performance is marginally better than that of XenServer. KVM falls slightly lower than other two hypervisors.

Network send / receive performance as long objects are captured with the below code snippet using SIGAR API. long receive = sigar.getNetInterfaceStat(sigar.getNetInterfaceList()).getRxBytes(); long sent = sigar.getNetInterfaceStat(sigar.getNetInterfaceList()). getTxBytes(); [Object sigar is an instance of SIGAR class]

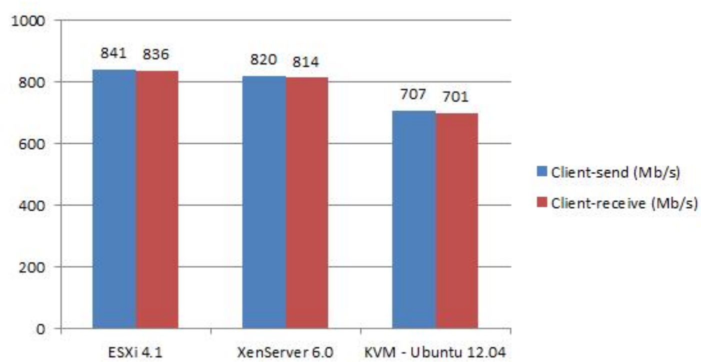


Figure. 5: Network Send-Receive performance captured using SIGAR (Higher values are better).

In Network Send-Receive performance experiment, in client-send test ESXi records 841 Mb/s speed, XenServer records 820 Mb/s speed and KVM records as 707 Mb/s speed with load generation on the three virtual machines. In client-receive test ESXi records 836 Mb/s speed, XenServer records 814 Mb/s speed and KVM records as 701 Mb/s speed with load tests are on. Under normal conditions hypervisors records Network speed of 941 Mb/s send and 939 Mb/s receive for given hardware configuration but with generated loads on three virtual machines hypervisors network performance has come down 11 to 12% of their original capacity. As the load is same on all three hypervisors and for generated stress, ESXi exhibits better network performance compared to other two hypervisors.

ESXi hypervisor exhibits high network throughput performance compared to other two hypervisors. XenServer hypervisors network throughput performance is less than ESXi and more than that of KVM. Hence, XenServer gives better network throughput performance compared to KVM but less than ESXi.

Due to virtualization overhead hypervisors need to handle more CPU instructions which eventually impact their network performance. Among three hypervisors, ESXis full virtualization technique gives high network throughput with low CPU utilization compared to other two hypervisors. KVM falls behind two hypervisors and needs improvement from network throughput performance perspective. Because of XenServers virtualization technique, it has to handle more CPU instructions hence it has impacted its CPU performance along with network throughput performance compared to ESXi. Hence for network intensive applications, ESXi is a best-suited hypervisor.

5. CONCLUSION

The intent of this paper is to evaluate the system resources performance of three hypervisors, namely VMWare ESXi Server, XenServer, and KVM for consolidated application workloads in the private cloud environment and recommend the hypervisors suitability for respective system resources intensive application workloads. Three virtual machines with three applications web server, application server, and the database server are deployed on each hypervisor in the private cloud. We have created a private cloud for the experiment. Load is generated on respective applications using open source load generating tools. Httperf is used to create load on the web server, JMeter is used to create load on the application server and Mysqlslap is used to create load on the database server. Once the entire experiment setup is ready then the load is generated concurrently by three applications on each hypervisor. System resource information is gathered with a java program using SIGAR framework.

Among three hypervisors, for the consolidated application workloads, VMWares ESXi shows better performance in CPU utilization and in network performance compared to other two hypervisors. XenServer shows better performance in memory utilization and in disk I/O performance compared to other two hypervisors. KVM needs to improve in all four system resources performance point of view.

Due to virtualization overhead hypervisors need to handle more CPU instructions compared to the native machine. Because of XenServers virtualization technique, it has to handle more CPU instructions hence it has impacted its CPU performance along with network throughput performance compared to ESXi. Hence for CPU and network throughput intensive applications, ESXi is a best-suited hypervisor. Because of ESXis virtualization technique, it has an overhead of trapping, scanning and translation which eventually impacted its memory and disk I/O performance compared to XenServer. Hence for memory and disk I/O intensive applications, XenServer is a best-suited hypervisor. Overall KVM falls behind two hypervisors with its virtualization technique, virtual device drivers slow performance and needs improvement on all four system resources performance perspective.

From the industry perspective, CPU and network intensive applications should use ESXi hypervisor. Gaming servers and Animation servers require high CPU intensity hence ESXi is recommended as a best-suited hypervisor for them. Application servers and Email servers require high network intensity hence ESXi is a best choice for these applications. Memory and disk I/O intensive applications should use XenServer hypervisor. Web servers require high memory intensity hence XenServer should be an apt choice here. DB servers require high disk I/O intensity therefore XenServer is most suited hypervisor for database applications.

Hypervisors recommendations for respective system resources intensive application workloads are depicted in Table III.

Table III: RECOMMENDATION OF HYPERVISORS FOR SYSTEM RESOURCE INTENSIVE APPLICATION WORKLOADS

Hypervisors	CPU Intensive Application workloads	Memory Intensive Application workloads	Disk I/O Intensive Application workloads	Network Intensive Application workloads
ESXi 4.1	Most Recommended	Recommended	Recommended	Most Recommended
XenServer 6.0	Recommended	Most Recommended	Most Recommended	Recommended
KVM - Ubuntu 12.04	Moderately Recommended	Moderately Recommended	Moderately Recommended	Moderately Recommended

Experimentation setup is creative and collecting information through SIGAR API on concurrent VMs is an interesting idea. Setting up three hypervisors as hosts in three different clusters (as CloudStack allows only homogeneous hypervisors in the cluster) and running three applications with VMs concurrently and collecting data simultaneously from all three VMs is a challenging task. From the results ESXi hypervisor, which uses full virtualization technique exhibits better performance in CPU utilization and Network speed. XenServer, a para-virtualized hypervisor exhibits better performance in memory utilization and disk I/O speed. On the other hand, KVM hypervisor needs to improve compared to other two hypervisors.

6. FUTURE WORK

Overall three hypervisors perform close to each other in all tests for the consolidated application workloads. ESXi scores over two hypervisors in CPU and network performance and XenServer scores over two hypervisors in memory and disk I/O due to their respective virtualization techniques that are used in their development.

Consolidated load environment and hardware configuration are same for all three hypervisors hence the hypervisors which are lacking in the respective system resources performance need an improvement. As a future work, we would like work on getting KVMs performance close to other two hypervisors by suggesting improvements in virtual device drivers development. Researchers may concentrate on the loopholes exhibited in the respective system resources performance tests by ESXi and XenServer and may suggest further improvements. The accurate system information indicates the gap which should be filled for improvement of hypervisors.

Cloud computing is evolving at a rapid pace hence more research is required in this area and as a future work more hypervisors can be evaluated from the performance perspective in the public and private cloud environments.

REFERENCES

Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., and Warfield, A. 2003. Xen and the art of virtualization. In *Proceedings of the Nineteenth ACM Symposium on Operating systems Principles (New York, 2003). ACM Press, New York, 164177.*

Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., and 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.*

FUJITSU. 2010. Benchmark Overview-vServCon.
FUJITSU. 2010. Benchmark Overview-vServCon. Last Accessed March, 2010.

Fujitsu Technology Solutions. 2007. DataSheet Citrix XenServer.
Fujitsu Technology Solutions. 2007. DataSheet Citrix XenServer. 2007.

Hostway UK. 2011. VMware ESXi Cloud Simplified. Comprehensive explanation of the features and benefits of VMware ESXi Hypervisor.
http://www.hostway.co.uk/small-business/dedicated-hosting/cloud/vmware-esxi.php. 2011.

Jianhua, C., Qinming, H., Qinghua, G., and Dawei, H. 2008. Performance Measuring and Comparing of Virtual Machine Monitors. Embedded and Ubiquitous Computing. In *EUC'08.* 2008.

MELL, P., AND GRANCE, T. 2009. The NIST Definition of Cloud Computing. In *National Institute of Standards and Technology, Information Technology Laboratory.* Version 15, 10-7-09.

MENON, A. 2005. Diagnosing Performance Overheads in the Xen Virtual Machine Environment. In *Conference on Virtual Execution Environments (VEE'05).* 2005

MOLLER, K.T. 2007. Virtual Machine Benchmarking. In *Diploma Thesis, Karlsruhe Institute of Technology, 2007.* 2007

NANDA, S., AND CHIUEH,T. 2005. A Survey on Virtualization Technologies. In *Technical report, Department of Computer Science, SUNY at Stony Brook. (New York, 2005.) 11794-4400.*

SIGAR. 2012.
https://support.hyperic.com/display/SIGAR/Home. 2012.

CLOUDSTACK. 2014.
http://cloudstack.apache.org. 2014.

MYSQLSLAP. 2009.
http://dev.mysql.com/doc/refman/5.1/en/mysqlslap.html. 2009.

HTTPERF. 2008.
http://www.hpl.hp.com/research/linux/httperf/. 2008.

JMETER. 2012.
http://jmeter.apache.org/. 2012.

NETPERF. 2012.
http://www.netperf.org/netperf/. 2012.

PASSMARK. 2014.
http://www.passmark.com/products/pt.htm. 2014.

SPEC. STANDARD PERFORMANCE EVALUATION CORPORATION. 2014.
http://http://www.spec.org/. 2014.

SPECVIRT_SC2010 DESIGN OVERVIEW.
http://www.spec.org/virt$_s$*c*2010/. last accessed on May 23, 2011.

REDDY V V, P., AND RAJAMANI,L. 2014. Performance Comparison of Hypervisors in the Private Cloud. In *Advanced Computing, Networking and Informatics,vol.2,2014.DOI:10.1007/978-3-319-07350-7_44.* *http://link.springer.com/chapter/*10.1007%2*F*978 − 3 − 319 − 07350 − 7$_4$4. 2014.

REDDY V V, P., AND RAJAMANI,L. 2014. Evaluation of Different Hypervisors Performance in the Private Cloud with SIGAR Framework. In *International Journal of Advanced Computer Science and Applications, Vol.5, No.2, 2014.* *http://dx.doi.org/*10.14569/*IJACSA*.2014.050210. 2014.

REDDY V V, P., AND RAJAMANI,L. 2014. Performance Evaluation of Hypervisors in the Private Cloud based on System Information using SIGAR Framework and for System Workloads using Passmark. In *International Journal of Advanced Science and Technology, Vol 70, 17-32.* 2014. *http://www.sersc.org/journals/IJAST/vol*70/3.*pdf*. 2014.

SHAN, Z., AND QINFEN, H. 2009. Network I/O Path Analysis in the Kernel-based Virtual Machine Environment through Tracing. In *Information Science and Engineering (ICISE).* 2009.

SRIDHARAN. SUGANYA. 2012. A Performance Comparison of Hypervisors for Cloud Computing. *http://digitalcommons.unf.edu/etd/*269. 2012.

VMWARE. 2007. A Performance Comparison of Hypervisors VMware. White paper. last accessed on February 1, 2007.

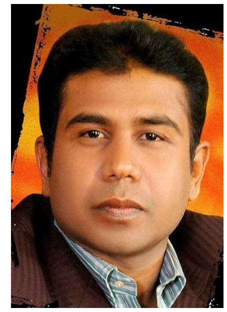
VMWARE. 2007. The Architecture of VMware ESXi. White paper. 2007.

VMWARE. 2007. Understanding Full Virtualization, Paravirtualization, and Hardware Assist. VMware, white paper. last accessed on November 10, 2007.

XEN ORGANIZATION. 2009. How does Xen work. 2009.

XENSOURCE. 2007. A Performance Comparison of Commercial Hypervisors. XenEnterprise vs. ESX Benchmark Results. 2007.

P. Vijaya Vardhan Reddy P. Vijaya Vardhan Reddy is an IEEE member. For the past 15 years, he has been working in the IT industry. He had worked in Tokyo (2001) for CSFB Project and in London (2005-06) for ADP Freedom Payroll Project in Java/J2EE Technologies. His recent stint was as an Assistant Vice President (AVP) at GE CAPITAL. He had actively involved in the design of web based multi-tier applications. He has experience in SDLC and project management methodologies like agile/scrum. He is a certified six sigma black belt professional. His research areas include distributed computing, grid computing and cloud computing.



LakshmiRajamani Lakshmi Rajamani is a retired professor. She has several papers published in journals across the world. She had served as a head of the department for computer science department from 2010 to 2012.