19th International Computer Science and Engineering Conference (ICSEC)
Chiang Mai, Thailand, 23-26 November, 2015

Performance Analysis of a Multi-Tier Video on Demand Service on Virtualization Platforms

Kritwara Rattanaopas and Pichaya Tandayya
Department of Computer Engineering,
Faculty of Engineering,
Prince of Songkla University,
Hatyai, Thailand
kritwara.ra@skru.ac.th, pichaya@coe.psu.ac.th

Abstract— **Cloud computing technology, especially virtualization is employed in many data centers nowadays. The key concept of virtualization concerns elastic or scalable infrastructure. This concept can be implemented by exploiting multi-tier web applications and hypervisors which are virtual machine management software. Xen hypervisor has presented its Version 4.4 in 2014. In this paper, we present the performance analysis comparison between Xen para-virtualization and KVM full virtualization on the case study of open source video streaming called Cumulusclips, a YouTube-like system. This investigation involves real workload mp4 video streaming on 200 clients' browser, running 3 experiments, including large video files (~3 GB), small video files (~120 MB) and random-size video files (the ratio of large and small video files is 25%/75%). The requests size results show that Xen para-virtualization can serve all requests better than KVM full virtualization and use less resource. Xen's performance is dropped when CPU usage is 100% in Experiment 1 (large files). In Experiments 2 (small files) and 3 (random-size files), Xen's CPU usage is under 10%, but KVM's CPU usage is over 50%. The requests size results of Xen and KVM are equal in Experiment 2, but Xen has the maximum throughput about a half (51%) of KVM's. Therefore, we can conclude that Xen para-virtualization has better performance than KVM full virtualization on multi-tier video streaming system.**

Keywords—Cloud computing; virtualization; performance comparison; Multi-tier applications; Video Streaming; Xen; KVM;

I. INTRODUCTION

Cloud computing [1] has become very popular. Many cloud services can be categorized into three types: Cloud Software (Software-as-a-Service), Cloud Platforms (Platform-as-a-Service) and Cloud Infrastructure (Infrastructure-as-a-Service). All services come in three forms: public clouds, private clouds, and hybrid clouds. The cloud service goals are to manage and allocate resources of data centers with a new type of services calls "as a service" [1] over the Internet. Cloud architectures have different infrastructures based on the of the hypervisor type employed. Hypervisor is a virtualization software, examples include VMware(ESXi), Hyper-V, Xen and KVM. Currently, Xen and KVM are the most popular open source virtualization software. Something as a service on the cloud is a major topic of research and its goal is to optimize and vary resources including CPU, memory and Input/output devices for dynamic users' requests, of which the infrastructure is called "Elastic Infrastructure" [2]. However, this keyword can lead to a lot of researches and new services on the cloud over the Internet nowadays.

In 2003, the new web application infrastructure called "multi-tier architecture" is presented. The first generation of multi-tier is deployed within only physical servers and proposed with a largest service as e-commerce [3],[4]. The e-commerce multi-tier web application includes a front-end web tier, a middle Java enterprise tier and a backend database tier that uses estimate about one physical server each tier to drive services and vary physical servers by using the dynamic response time of requests or sessions which is called Service Level Agreements (SLA). On the other hand, single-tier is an easy way to configure general web application service and control SLA back then.

Currently, the virtualization technology can provide many virtual machines on a physical server which has more than 16 cores and over 128 GB RAM. Amazon's Elastic Computing Cloud (EC2) [1] presented Infrastructure-as-a-Service (IaaS), a pay-as-you-go cloud computing environment [5]. Virtualization technology is compatible with multi-tier architecture on web application service and can allocate dynamic resources to virtual machines of each tier. Hence, research topics about multi-tier presented predictive models for web service workload and because each tier has significantly different requirements of physical server resources with tradeoff the quality of service (QoS).

In this paper, we evaluate the performance of two open source hypervisors, Xen (para-virtualization) and KVM (full virtualization) which have different virtual machines infrastructure to access physical resources. The new Xen version 4.4 launch in March 10, 2014 presented scope for improvement in its para-virtualized mode. The evaluation is done by measuring the performance of both hypervisor platforms with a video streaming service workload called Cumulusclips [6] which is a YouTube-like system. Video streaming services include Youtube, dailymotion, Vevo and vimeo have become very popular services and consume more resources for SLA. HTTP status code of streaming service is 206, which is different from 200, means the server splits a

978-1-4673-7825-3/15/$31.00

video into multiple small streams and records transactions to log after the end of a video. Only the 206 HTTP code is the most difficult to determine the cost of SLA and research in this topic is rare. A few recent researches are focused on the average response time or 200 HTTP code. Our goal is to compare the hypervisor performance with the workload of video streaming services.

The rest of the paper is organized as follows. In Section II, we describe related work and background review including multi-tier web application, virtualization technology (Xen and KVM) and Cumulus Clips. In Section III, we describe the research methodology including experimental setup and workloads. Section IV shows results evaluation and Section V discusses the experimental results. Finally, Section VI concludes and reveals possible future works.

II. Background Review and Related Work

A. Related Works

Several researches involved multi-tier web applications have previously focused on the issue of predicted SLA in data centers. In 2005, B. Urgaonkar et al. presented the model which was implemented using the Mean-Value Analysis (MVA) algorithm for response time's prediction with a real Linux servers cluster. Y. Diao et al. [7], using the same approach, purposed a tier-to-tier management architecture to control a multi-tier system using the IBM Web Sphere Application Server and DB2 using service level objectives (SLOs). W. Iqbal et al. [8] continued to research on automatic detection and demonstrated their approach by applying RUBiS [9] workload on a EUCALYPTUS cloud. This research could not provide a minimal resource over all multi-tier web applications. N. Grozev and R. Buyya [10] presented performance modeling of a multi-tier system by surveying single-tier approaches and continued to simulate the same workload on heterogeneous Multi-Cloud on CloudSim[11]. The performance comparison of KVM and Xen in full virtualization mode has been proposed in [12]. This research presented a new approach for performance comparison between KVM full virtualization and Xen para-virtualization applying a video streaming workload.

B. Multi-Tier Web Application

In principle, multi-tier architecture is difficult to setup than single-tier architecture which is simpler. Nowadays, most web sites use multi-tier architecture. However, in multi-tier architecture, the resource allocation and predicted SLA problems will be more difficult due to different requirements and resources on each tier. Multi-tier web application is a concept of elastic services which include web servers, database servers and file storage servers. Each service is called a Tier as shown in Fig. 1.

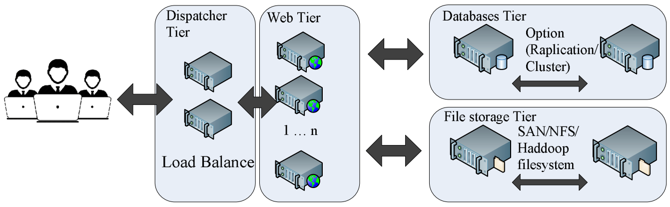


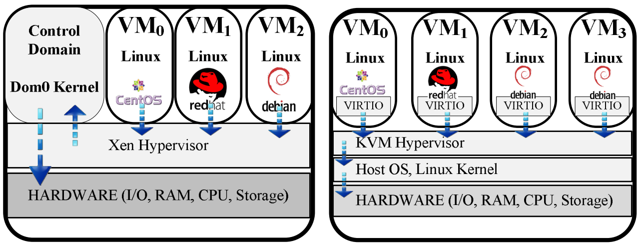
Fig. 1. Multi-Tier Architecture

For multi-tier architecture, it mainly has four tiers. The first tier named Dispatcher Tier is a frontend of multi-tier. The goal of this tier is to optimize or balance workloads for the web in the next tier. Examples of load balancer software [13] include Round-robin DNS, Rad Hat cluster suite's Piranha load balancing [14] and HAProxy [15]. The second tier named Web Tier consisting of web servers and web services. This tier is the center of other tiers. The third tier named Database Tier consists of database servers. Examples of database management software include Oracle, DB2, MySQL and PostgreSQL. The goal of this tier is to store and retrieve service information (e.g. user, service data, service category and report). The last tier named File Storage Tier consists of network storage servers, for example, popular network file servers [16] including storage area network (SAN), network file system (NFS) and Hadoop distributed file system (HDFS) [17]. The goal of this tier is to store and retrieve dynamic content files (e.g. document, multimedia and picture).

C. Virtualization

Virtualization technology provides a modern way to manage CPUs, memory, and other resources of physical host servers as cloud computing system infrastructure. The host virtualization software can be divided into two types: the physical host server OS type and the hypervisor type. The virtualization layer of the Host OS type called Guest OS is installed on top of Windows, Linux or other OS. The main concept of virtualization for this type is that a choice can be chosen between 1) separation of VMs and the physical host server, and 2) resources sharing.

Popular open source hypervisors, Xen and KVM, deploy modifications of the Linux kernel. KVM is based on a physical host server OS and adapts the kernel to a bare metal hypervisor in Fig. 2-b. Xen can deploy virtual machines including either full virtualization or para-virtualization platform with Dom0 (Domain Zero or Xen's "host" operating system) in Fig. 2-a. On the other hand, KVM has only a full virtualization platform. In full virtualization, the virtual machine must simulate hardware for Guest OS, but para-virtualization offers a para-API which runs directly on physical hardware. The architectures of Xen and KVM virtualization are illustrated in Fig. 2.



a) Xen b) KVM

Fig. 2. Hypervisor Architecture

D. Xen

The latest version of Xen hypervisor [18] since 10th March 2014 is 4.4. It can be installed on CentOS 6.6. This release improved disk and I/O drivers of fully Para-virtualization (PV) to perform better than Hardware-assisted Virtualization Mode (HVM). PV provides API used by virtual machines. On the other hand, HVM is Xen's full virtualization mode uses virtualization extensions from the host CPU to virtualize guests and requires Intel VT or AMD-V hardware extensions. HVM has new instructions to support direct calls by adding a para-virtualized guest/driver into the hypervisor, typically used for I/O. Xen uses Qemu to emulate physical hardware including CPUs, BIOS, local storage, USB controller, network adapter, etc. Therefore, the performance using HVM will be better than the full virtualization.

E. KVM

KVM based virtualization [19] is an open source full virtualization technology for Linux. It also supports Intel VT or AMD-V. Currently, the kernel component of KVM is included in mainline upper Linux 2.6.20 with QEMU 1.3 and it has VIRTIO to supports I/O resource usage. It can allocate guest's memory over-commitment of host's memory. New feature of KVM is the Disk Cache mode for improved I/O performance.

F. Cumulus Clips

Cumulus Clips is a free video sharing CMS like YouTube. It can provide videos on workstations and mobile devices which include iOS and Android, as well as all major browsers, including Firefox, Safari, Chrome, Opera, IE7+, that support high definition movies using the mp4 format. It can be used in a networked server for designing and testing workload performance in forms of video on demand (VOD) streaming spread used on YouTube and Daily Motion.

III. METHODOLOGY

In this section, we present the testing scenarios for investigating on the performance of the multi-tier architecture for VOD workloads and determining a suitable hypervisor for the service.

A. Experimental Setup

Employing para-virtualization (PV) of Xen and full virtualization (FV) of KVM, we experiment with Multi-Tier over two hypervisor platforms on a rack server with Intel Xeon E5630 Quad core @ 2.53GHz CPUs with support for Intel-VT, 8 GB DDR3 ECC Registered RAM @ 1600 MHz, 1.0 TB SATA disks and Intel® 82576EB Gigabit Ethernet Controller. Our system architecture for the testing environment is described as follows:

Multi-Tier: Infrastructure includes Web, Database and File storage Tiers. Tier details are described as follows:

- *Web server* involves 3 VMs including 1 vCPU, 1.5 GB RAM and 30 GB disk storage and software components (e.g. Cumulusclips and Apache HTTP server).
- *Database server* employs 1 VM including 1 vCPU, 0.5 GB RAM and 50 GB disk storage and software components (e.g. MySQL server and Apache HTTP server and Adminer Database management).
- *File storage server* exploits 1 VM including 1 vCPU, 0.5 GB RAM and 50 GB disk storage and software components (e.g. NFS server).
- *Load balance* is serviced by Round-robin DNS technique which is only returned IP addresses of several servers. This service excludes a performance model.

Host OS: Two hypervisor platforms run on CentOS 6.6. The hypervisors and all tiers of VMs mentioned above without Load balancing have been installed on separate disk partitions of the same hard disk. The VMs were stored on the XFS format partition.

Hypervisor Versions: qemu-kvm-0.12.1.2 and new xen-hypervisor-4.4. We installed Syslog-ng for measurement report about CPU and memory usages, which employed Perl scripts to transferring them into a MySQL database format.

B. Workload

The characteristic of video streaming workload has a major effect on input and output devices including disk storage and network [20]. We use Cumulus Clips which is an open-source video sharing application in our experimental study to compare KVM's full virtualization platform and Xen's para-virtualization platform. Video workload is implemented by two types of files including small files and large files.

- Workload of small files has 40 720p-resolution files of which duration of video is about 7-9 minutes.
- Workload of Large files has ten 720p-1080p resolution files of which duration of video is about 60 minute.

We performed all experiments based on these video workloads. The video files were opened via client web browser following a sequence with a 10-second gap between each adjacent pair, starting from 1 to 200 web browser(four web browsers per computer) with three scenarios including large files, small files and random files (the ratio of large files and small file is 25%/75%).

IV. EVALUATION

This section reports the results on resource usage of video streaming over a multi-tier web application. The results of 3 experiments show relation between response time of 206

HTTP code with requested size and network throughput with CPU usage for each hypervisor.

A. Experiment 1 : large files workload

In Experiment 1, we opened only large video files on browser. The average size of large files is nearly 3.0 GB. We stored the resources usage data using syslog-ng and perl scripts. The goal of this experiment is to show the heavy workload of video streaming in the multi-tier web system (Fig. 3). We can describe the resource management in Fig. 4 and Fig. 5.

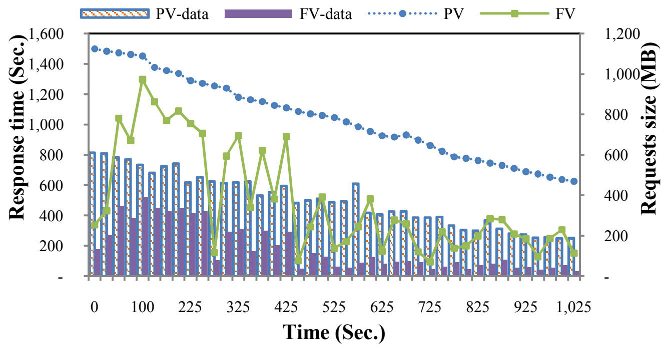


Fig. 3. Comparison of Hypervisors' Relationship between Response Time and Requests Size Applying Large Files Workload

A bar graph in Fig. 3 shows the decrease of requests size from 612 to 189 MB in Xen para-virtualization (PV-data), meanwhile, the maximum requests size a bar graph of KVM full virtualization (FV-data) is only 390 MB. A response time trend of Xen para-virtualization (PV) is more linear than that of KVM full virtualization (FV). Although the response time decreases with time, themulti-tier video streaming system using either hypervisor is unable to serve all requests in Experiment 1.

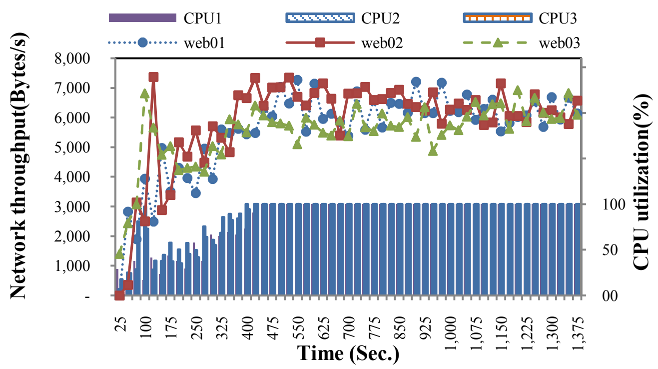


Fig. 4. Relationship between Network Throughput and CPU Utilization for KVM Full Virtualization Virtual Machine Applying Large Files Workload

In Fig. 4 shows the full CPU utilization of three KVM virtual machines and that the throughput is more than 7,000 bytes/s on each web server on Tier-1. This throughput remains nearly constant at 6,000 bytes/s although the CPU is fully utilized during Experiment 1.

In this experiment, Fig. 5 shows the throughputs of three Xen virtual machines decrease with time at 100% CPU utilization. Throughput is lower than 4,000 bytes/s. Therefore, the throughput of multi-tier video streaming system with Xen para-virtualization is unable to serve all requests at full CPU utilization.

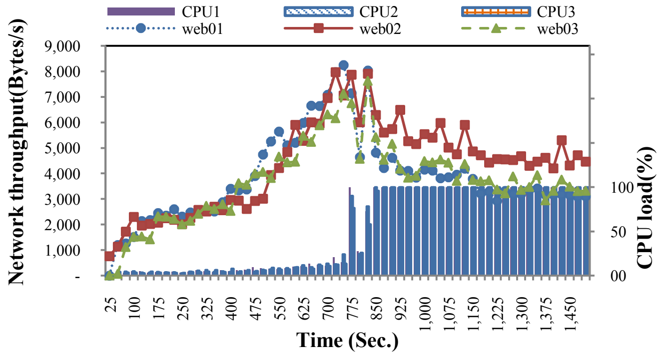


Fig. 5. Relationship between Network Throughput and CPU utilization of Xen Para-virtualization Virtual Machine Applying Large Files Workload

B. Experiment 2 : small files workload

In Experiment 2, we opened only small video files on 200 browsers. In the bar graph, the average size of the small files workload is nearly 121 MB. The experiment shows the light weight workload of video streaming in the multi-tier web system on both of Xen para-virtualization (PV-data) and KVM full virtualization (FV-data) in Fig. 6.

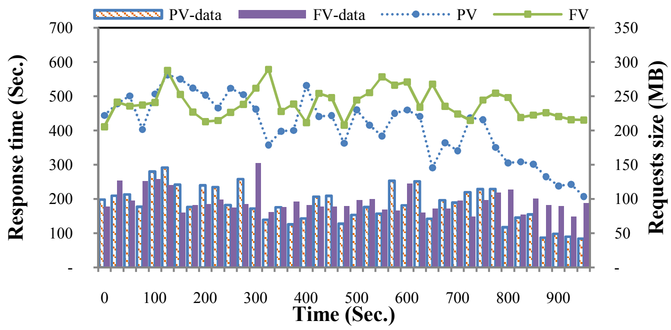


Fig. 6. Comparison of Hypervisors' Relationship between Response Time and Requests Size Applying Small Files Workload

Fig. 6 shows that both of Xen para-virtualization (PV) and KVM full virtualization (FV) have similar performance. The response time is lower than 600 seconds. Therefore, the multi-tier video streaming system employing either hypervisor can serve all requests in Experiment 2.

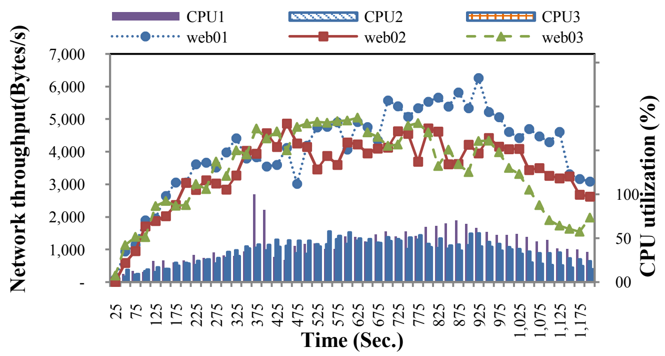


Fig. 7. Relationship between Network Throughput and CPU Utilization for KVM Full Virtualization Virtual Machine Applying Small Files Workload

Fig. 7 shows the throughput of three KVM virtual machines varies from 3,000 to 5,800 Bytes/s and the CPU usage is around 50% on each web server during Experiment 2.

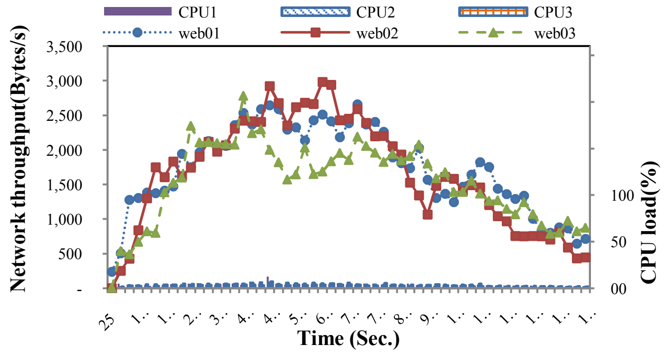


Fig. 8. Relationship between Network Throughput and CPU utilization of Xen Para-virtualization Virtual Machine Applying Small Files Workload

The throughput of three Xen virtual machines in Fig. 8 is only half of KVM virtual machines for the throughput between 1,480 – 2,980 Byte/s. The CPU usage is only 5%, a 10-fold-lower than the CPU utilization of KVM virtual machine. For this reason, the information about resource usage in video streaming service shows that Xen 4.4 para-virtuaizatuion provides better performance than KVM full virtualization.

C. Experiment 3 : random files workload

In Experiment 3, we randomly opened files, one large and three small video files, on browser. The experiment is to show the real workload of video streaming in the multi-tier web system (Fig. 9). Therefore, the multi-tier video streaming system using either hypervisor is unable to serve all requests as it has similar performance comparing to Experiment 1.

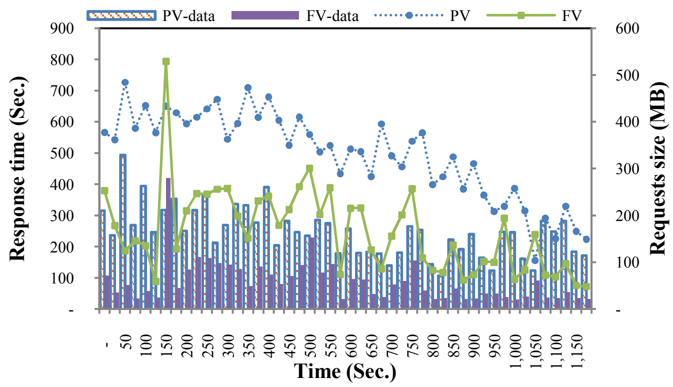


Fig. 9. Comparison of Hypervisors' Relationship between Response Time and Requests Size Applying Random Files Workload

Fig. 9 shows the different performances from both hypervisors. Clearly, the requests size of Xen para-virtualization (PV-data) is higher than KVM full virtualization (FV-data) during the experiment. KVM full virtualization decreases the requests size after the 500^{th} second, but Xen virtual machine can serve all requests. The requests size in a bar graph of Xen para-virtualization (PV-data) runs over 100 MB during the experiment.

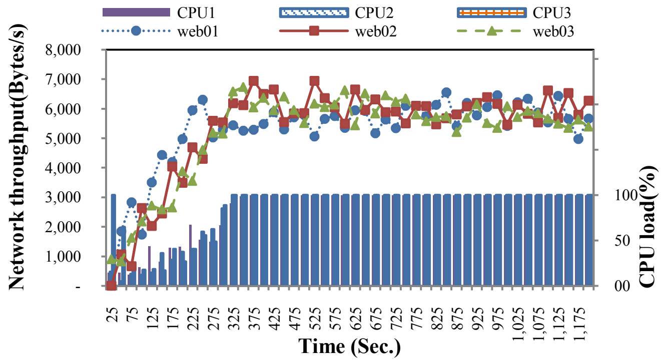


Fig. 10. Relationship between Network Throughput and CPU Utilization for KVM Full Virtualization Virtual Machine Applying Random Files Workload

Fig. 10 shows the throughput of three KVM virtual machines remains nearly constant at 6,000 bytes/s and the CPU usage is 100% on each web server, similar to Experiment 1. The results imply that this system is unable to serve all requests of the multi-tier web system.

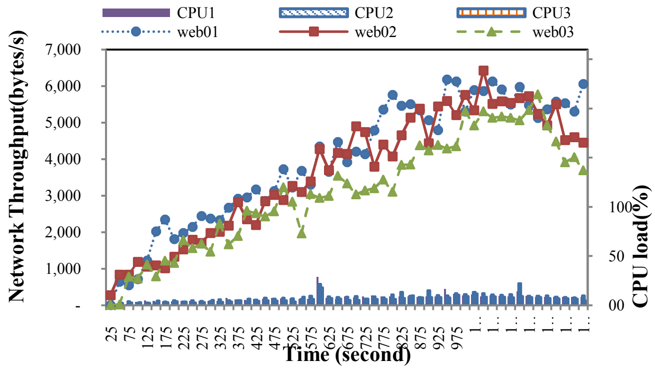


Fig. 11. Relationship between Network Throughput and CPU utilization of Xen Para-virtualization Virtual Machine Applying Random Files Workload

Fig. 11 shows the maximum throughput of three Xen virtual machines is at 6,426 bytes/s and the CPU usage is under 10% during Experiment 3. Similarly, results of this experiment shows that Xen 4.4 can provide better performance than KVM.

V. Results and Discussion

We have investigated a multi-tier web application applying two new virtualization open source technologies for resources management with real video streaming service workload and conducted three experiments including large files, small files and random-size files. The results can be concluded as follows.

In Experiment 1, the workload contains only large video files for streaming. The average file size is 3 GB. Therefore, a multi-tier video streaming system applying Xen para-virtualization and KVM full virtualization are unable to serve all requests. The maximum throughput of both hypervisors is around 7,000 Bytes/s and the CPU usage is 100%. If Xen virtual machine's CPU utilization is 100% then the overall performance will be dropped. On the other hand, KVM virtual machines still maintains the maximum throughput after the CPU usage is 100%. The requests size result shows that Xen has better performance than KVM.

In Experiment 2, the workload consists of only small files for streaming. The average size is 120 MB. Therefore, a multi-tier video streaming system applying Xen para-virtualization and KVM full virtualization can serve all requests during the experiment. The maximum throughput of KVM is around 6,000 Bytes/s and the CPU usage is around 50%, but Xen's maximum throughput is only 2,980 Byte/s and the CPU usage is under 5%. Clearly, Xen uses fewer resources than KVM.

In Experiment 3, large files and small files with the ratio of 25%/75% are used for testing streaming workload. KVM's performance is similar to the results in Experiment 1, but Xen's performance was enough until the 1,025th second and the server requests size is over 100 MB during the experiment. Therefore, in summary, Xen para-virtualization has better performance than KVM full virtualization.

VI. CONCLUSIONS

In conclusion, in Xen version 4.4 with para-virtualization, the CPU usage is as low as 10% which is about 10 folds comparing to KVM full virtualization. That is similar to the results of Experiments 2 and 3. CPU usage is a necessary factor in the power consumption problem as the power consumption calculation can be shown in Equation (1) where P_{server} is the estimated power consumption of the physical server, P_{CPU} is the CPU power consumption, P_{mem} is the memory power consumption, P_{io} is the disk I/O power consumption and P_{idle} is the power consumed by an idle server.

$$P_{server} = P_{CPU} + P_{mem} + P_{io} + P_{idle} \quad (1)$$

Our experiment results also show that Xen's CPU usage is lower than that of KVM. However, the server's power consumption increases as the CPU usage increases. Therefore, we expect that the power consumption of Xen virtual machine should be less than that of KVM virtual machine. Our finding proves that in order to improve the performance and power consumption of a multimedia video streaming system, the choice of hypervisor type used plays an important role.

In future work, we will propose an autonomic system which combines Xen and KVM hypervisors into a multi-hypervisor environment for optimizing the performance meanwhile reducing the power consumption.

ACKNOWLEDGMENT

The authors would like to thank Prince of Songkla University for the Graduate Study Scholarship and Songkhla Rajabhat University for the computing and storage resources.

REFERENCES

[1] C. Landis and D. Blacharski, *"CLOUD COMPUTING MADE EASY,"* March 16, 2013.

[2] C. Fehling, F. Leymann, R. Retter, W. Schupeck and P. Arbitter, *" Cloud Computing Patterns, "* 2014, [E-book] Available: http://www .cloudcomputingpatterns.org/Cloud_Computing_Patterns.

[3] A. Beloglazov, R. Buyya, Y. C. Lee and A. Zomaya, "A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems", *arXiv:1007.0066 [cs]*, July 2010.

[4] B. Urgaonkar, G. Pacifici, P. Shenoy, M. Spreitzer and A. Tantawi, "An analytical model for multi-tier internet services and its applications", in *In Proc. of the ACM SIGMETRICS'2005*, 2005, pp. 291–302.

[5] D. Huang, B. He and C. Miao, "A Survey of Resource Management in Multi-Tier Web Applications", *IEEE Communications Surveys Tutorials*, 16th, vol. 3, Third 2014, pp. 1574–1590.

[6] cumulusclips.org, *"Cumulusclips, "* 2011. http://cumulusclips.org/.

[7] Y. Diao, J. L. Hellerstein, S. Parekh, H. Shaikh and M. Surendra, "Controlling Quality of Service in Multi-Tier Web Applications", in *26th IEEE International Conference on Distributed Computing Systems, 2006. ICDCS 2006*, 2006, pp. 25–25.

[8] W. Iqbal, M. N. Dailey, D. Carrera and P. Janecek, "Adaptive resource provisioning for read intensive multi-tier applications in the cloud", *Future Generation Computer Systems*, 27th , vol. 6, 2011, pp. 871–879.

[9] OW2 consortium, *" RUBiS: an auction site prototype, "* , 1999. http://rubis.ow2.org/.

[10] N. Grozev and R. Buyya, "Performance Modelling and Simulation of Three-Tier Applications in Cloud and Multi-Cloud Environments", *The Computer Journal*, 58th , vol. 1, Jan 2015, pp. 1–22.

[11] R. N. Calheiros, R. Ranjan, A. Beloglazov, C.A.F.D. Rose and R. Buyya, "CloudSim : a tool kit for mode lingand simulation of cloudcomputing environments and evaluation of resource provisioning algorithms", *Software : Practiceand Experience*, 41st , 2011, pp.23–50.

[12] K. Rattanaopas and P. Tandayya, "Comparison of Disk I/O Power Consumption in Modern Virtualization", in *International Conference on Computer, Communication, and Control Technology (I4CT 2015)*, 2015, pp. 49–53 .

[13] "Load balancing (computing), " in Wikipedia, the Free Encyclopedia [Online], July 28, 2015. Available: http://en.wikipedia.org/wiki /Load_balancing_(computing) [July 30, 2015].

[14] "Red Hat cluster suite, " in Wikipedia, the Free Encyclopedia [Online], July 21, 2015. Available: https://en.wikipedia.org/wiki/Red_ Hat_cluster_suite [July 30, 2015].

[15] "HAProxy, " in Wikipedia, the Free Encyclopedia [Online], July 30, 2015. Available: http://en.wikipedia.org/wiki/HAProxy [July 30, 2015].

[16] "Storage area network, " in Wikipedia, the Free Encyclopedia [Online], July 9, 2015. Available: http://en.wikipedia.org/wiki/Storage _area_network [July 30, 2015].

[17] D. Borthakur, *" HDFS Architecture Guide, "* 2011. http://hadoop. apache.org/docs/r1.2.1 /hdfs_design.html.

[18] "Xen, " in Wikipedia, the Free Encyclopedia [Online], September 28, 2014. Available: http://en.wikipedia.org/wiki/Xen [October 10, 2014].

[19] Red Hat, *"Kvm - kernel based virtual machine,"* Technical report, 2009.

[20] P. Pegus II, E. Cecchet and P. Shenoy, "Video BenchLab: An Open Platform for Realistic Benchmarking of Streaming Media Workloads", in *Proceedings of the 6th ACM Multimedia Systems Conference*, New York, NY, USA, 2015, pp. 165–176.