International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:11, No:3, 2017

Operating System Based Virtualization Models in Cloud Computing

Dev Ras Pandey, Bharat Mishra, S. K. Tripathi

***Abstract*—**Cloud computing is ready to transform the structure of businesses and learning through supplying the real-time applications and provide an immediate help for small to medium sized businesses. The ability to run a hypervisor inside a virtual machine is important feature of virtualization and it is called nested virtualization. In today's growing field of information technology, many of the virtualization models are available, that provide a convenient approach to implement, but decision for a single model selection is difficult. This paper explains the applications of operating system based virtualization in cloud computing with an appropriate/suitable model with their different specifications and user's requirements. In the present paper, most popular models are selected, and the selection was based on container and hypervisor based virtualization. Selected models were compared with a wide range of user's requirements as number of CPUs, memory size, nested virtualization supports, live migration and commercial supports, etc. and we identified a most suitable model of virtualization.

***Keywords*—**Virtualization, OS based virtualization, container and hypervisor based virtualization.

I. Introduction

CLOUD computing holds the potential to eliminate the requirements of setting up of high-cost computing infrastructure for the IT-based solutions and services. Virtualization in computing is creation of something virtual such as hardware, software, platform, storage, or network devices. Hardware and networking resources in a single hosting environment are basically abstracted and isolated. Virtualization increases the security of cloud computing and also integrates guest virtual machines and also cloud computing with the help of virtual machines through scale up and down on demand and as well as provides reliability [1], [9]. We should virtualize because of isolation among user resource sharing, dynamical resources, dynamical resources, and aggregation of resources [2], [10]. Today, it is believed that many of the organization are not feeling confident and secured to move on cloud because of its certain excuses of its architecture which made cloud computing most vulnerable of various security and threats [6].

It is found that operating system (OS) based virtualization is an easier and suitable approach. Application based virtualization is found to more memory consuming because of abstraction of host OS, whereas hypervisor based virtualization causes a single point of failure, and taking control over the hypervisor from virtual machine level is not possible [3]. OS based virtualization model is also categorized as container based and hypervisor based virtualization. Under this, a wide group of models is available with their specification, which makes difficult to select one. On the basis of user's requirements, different models of OS based virtualization and both type of techniques container based and hypervisor based are selected for further study incorporated with requirements of users.

Dev Ras Pandey (Research Scholar) is Mahatma Gandhi Chitrakoot Gramodaya Vishwavidyalya, Chitrakoot, Satna (M.P.), India (e-mail: devraspandey@gmail.com).

Bharat Mishra (Associate Professor) and S. K. Tripathi (Associate Professor) are with the, Mahatma Gandhi Chitrakoot Gramodaya Vishwavidyalya, Chitrakoot, Satna (M.P.), India.

II. OS Based Virtualization

Host OS enables virtualization that supports multiple isolated, virtualized guests – OS instances, nested virtualization and commercial supports etc. on a single physical server whose OS kernel is shared by many machines. OS virtualization takes advantage of the architectural design of OS [4]. Kernel of the host OS is shared by the virtual guest OS, whereas all the virtual machines have their own root files system. This structure is illustrated in Fig. 1. Partition on a single physical server and the use of OS instances is responsible for work independently in each guest application environment [5], [11]. The technique where the kernel of an OS allows isolation of multiple user space instances is known as server virtualization. The instance runs on top of pre-existing host OS, and a set of libraries is provided to interact with applications with giving them the warning that they are running on a single machine to its use. The dedicated instances are known as container, virtual server or virtual environment, or virtual machines [12].

For deploying and running distributed applications without launching an entire virtual machine for each application, container based virtualization method is implemented. As an alternative, multiple isolated systems are treated as containers that run on a single control host and access a single kernel, whereas hypervisor is a piece of software, firmware, or hardware that creates and runs virtual machines. Because containers share a host OS, security threats have easier access to the entire system when compared with hypervisor based virtualization. Another one is that it uses only same OS as the base OS, whereas hypervisor instances can run each unique OS [7].

With the OS based virtualization, we install the software on the top of the guest OS on host OS, but with each guest, OS have its own resources and are running in complete isolation from the other guest OS as OS based virtualization models.

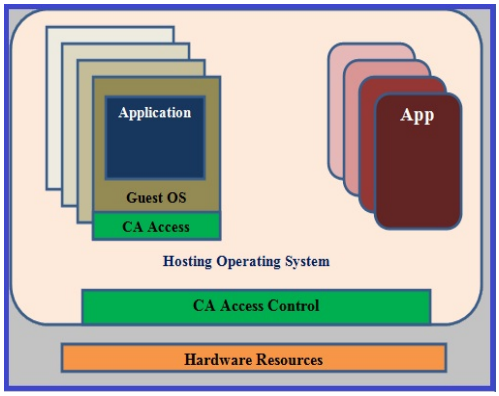

Fig. 1 OS based virtualization

III. OS Based Virtualization Models

Both containers based and hypervisor based virtualization models were compared with the various parameters. File system isolation is a data storage mechanism that provides isolation and safety by defining standardized ways of associating code with saved data. Disk quota is a limit set by a system administration that restricts certain aspects of file system usage on modern OS. The function of disk quota is to allocate limited disk space in a reasonable way. A limit on input/output rate defines the application swapping and rate of transfer of data during running the virtualization. Memory limits define the total minimum and maximum memory used or recommended to use in it, whereas CPU quota determines maximum numbers of CPUs used in it. Network isolation means to enable network as multiple identical copies or clones of the environment. The term nested virtualization defines the ability to run a hypervisor inside of a virtual machine. The feature that allows running hypervisor inside virtual machine or on hypervisor is associated with nested virtualization. Partition check pointing and live migration can save the full state of running virtual environment to restore it later on the same or on a different host in a way transparent to running applications and network connections. Root privilege isolation refers to the permissions regarding root files of the OS.

TABLE I
Proprietary Licensed Virtualization Models Comparison Based on Various Parameters [8], [13]

SN	Model	OS	I/O rate limiting	Memory limits	CPU quotas	Network isolation	Nested virtualization	Partition check pointing and live migration	Root privilege isolation	File system isolation	Disk quotas	USB Support	Commercial Support
1	**Parallels Workstation**	Windows XP, Windows Vista and Windows Server 2003	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes
2	**PearPC**	FreeBSD, Linux, Windows	Yes	Yes	Yes	No	No	Yes	Yes	Yes	Yes	No	No
3	**QEMU**	Linux, Windows, MAC OS X, UNIX	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	No	No
4	**Sandboxie**	Windows	No	No	No	-	No	No	Yes	Yes	-	No	No
5	**Spoon**	Windows	Yes	Yes	No	Yes	No	No	Yes	Yes	No	No	Yes
6	**VirtualBox**	Windows, Linux	Yes	Yes	Yes	Yes	No	No	No	Yes	Yes	Yes	Yes
7	**Virtuozzo**	Linux, Windows	Yes	Yes	Yes	Yes	-	Yes	Yes	No	Yes	Yes	No
8	**VMware ESXi**	Windows	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
9	**VMware Fusion**	MAC OS X	Yes	Yes	Yes	No	No	No	Yes	Yes	No	Yes	Yes
10	**VMware ThinApp**	Windows	No	No	No	Yes	No	No	Yes	Yes	No	No	No

TABLE II
Freeware Licensed Virtualization Models Comparison Based on Various Parameters [8], [13]

SN	Model	OS	I/O rate limiting	Memory limits	CPU quotas	Network isolation	Nested virtualization	Partition check pointing and live migration	Root privilege isolation	File system isolation	Disk quotas	USB Support	Commercial Support
1	**Hyper-V**	Windows	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
2	**iCore Virtual Accounts**	Windows XP	No	No	No	No	-	No	-	Yes	Yes	No	No
3	**Microsoft Virtual Server**	Windows XP, Windows Vista and Windows Server 2003	No	No	Yes	Yes	No	No	Yes	Yes	Yes	No	No
4	**VMware Server**	Windows, Linux	No	No	Yes	Yes	Yes	Yes	Yes	No	No	Yes	Yes
5	**VMware Workstation**	Windows, Linux	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
6	**VMware Workstation Player**	Windows, Linux	No	No	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes
7	**Windows Virtual PC**	Windows – 7	Yes	Yes	No	Yes	No	Yes	No	Yes	Yes	Yes	No

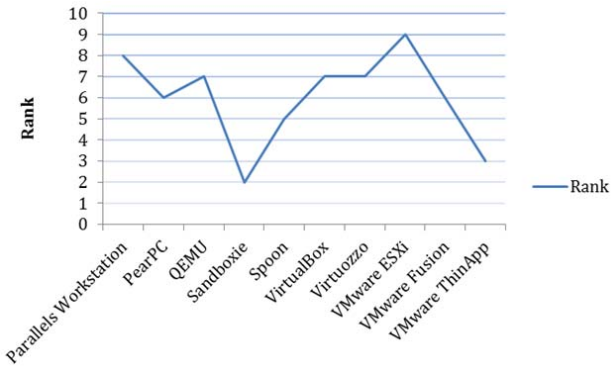

Fig. 2 Proprietary models comparison with their rank

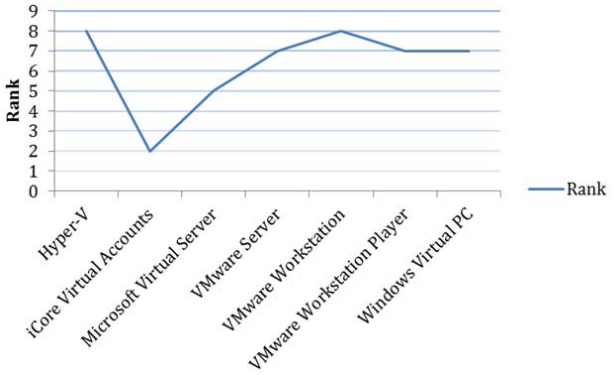

Fig. 3 Freeware models comparison with their rank

Here, many of models are available but the most popular models are selected for the study. Tables I and II define the properties of different virtualization models. Table I is selected as proprietary based licensed models of virtualization and Table II as freeware licensed models of virtualization. The values given in table with its associated attributes are in the form of yes or no. The comparison graph is drawn assuming that yes value is 1 and no value is 0, whereas the third value partial is also collected in the study and assuming 0.5 as shown in Fig. 2. These values are calculated as sum of their attributes and noted as rank of each model. On the basis of their rank, models were identified as suitable or not. In both Tables I and II, most of the parameters are noted as yes value, but in the case of first three parameters, i.e. I/O rate limiting, memory limits, and CPU quotas, if the values are yes, then it has limits and if the values are no then it has no limits, which is more preferable.

IV. Result and Discussion

In this study, many of the models with their own specification are selected as container based and hypervisor based models with supported platform of Windows. The maximum response of our specification and technology used models are VMware ESXi in proprietary based and in freeware Hyper – V and VMware Workstations that support multiple processors, memory and as well as nested virtualization. In freeware models, Hyper – V and VMware workstations do not support limits on input/output, memory and CPU but VMware workstation has a limit on File system and Disk Quota, which is a drawback and is shown in Table II. Fig. 2 explains the comparison of proprietary based models of virtualization. VMware ESXi has maximum value of its rank that shows it best. Fig. 3 shows two models with higher rank in freeware virtualization models; namely, Hyper – V and VMware Workstation. Fig. 4 describes overall comparison of proprietary based and freeware based virtualization models. VMware ESXi holds maximum rank, but it is an expensive model, so the freeware model is suggested to use. Hyper – V, a freeware OS based virtualization model has no limit on I/O rate, memory and limits on number of CPUs.

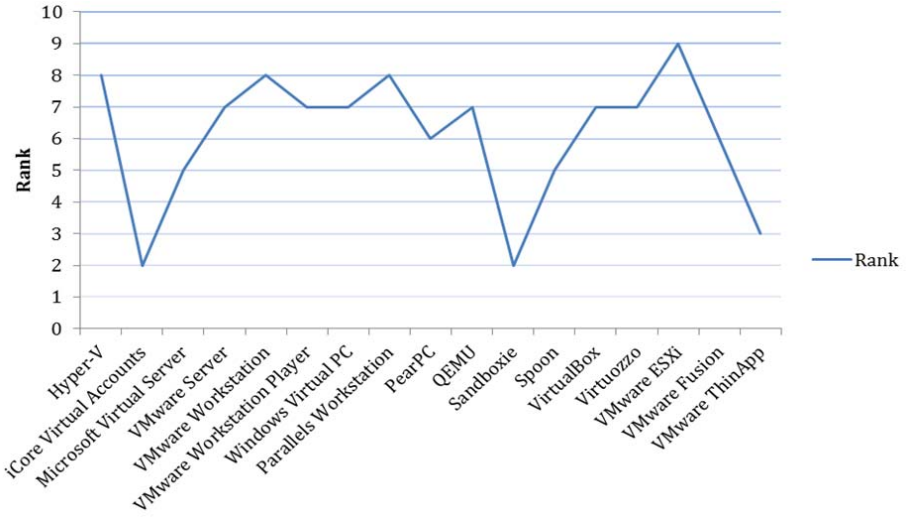

Fig. 4 Comparison of different models with their rank

V. Conclusion

Virtualization in computing is the creation of virtual of virtual something such as hardware, software, platform, storage or network devices. VMware ESXi, Hyper – V, and VMware workstations are found suitable because it supports multiple processors, memory and as well as nested virtualization. Hyper – V and VMware workstations do not supports limits on input/output, memory, and CPU, but VMware Workstation has a limit of disk quota. Because of a limit in VMware Workstation, Hyper – V is suggested to use as OS based virtualization model in cloud computing.

References

[1] T. Swathi, K. Srikanth and S. Raghunath Reddy, Virtualization in Cloud Computing, International Journal of Computer Science and Mobile Computing, Vol. 3, Issue 5, 540 – 546, 2014.
[2] Lakshay Malhotra, Devyani Agarwal and Arunima Jaiswal, Virtualization in Cloud Computing, Information Technology & Software Engineering, J Inform Tech Software Eng., 2014.
[3] Bharat Mishra, Dev Ras Pandey, Different approaches of virtualization in Cloud Computing, International Journal of Scientific Research, Volume 5, Issue 11, 672 – 673, 2016.
[4] Morty Eisen, Introduction to Virtualization, The Long Island Chapter of the IEEE Circuits and Systems (CAS) Society, 2011.
[5] Shikha R. Thakur, R. M. Goudar, Improving Network I/O Virtualization Performance of Xen Hypervisor, International Journal of Engineering Trends and Technology (IJETT) – Volume 11 Number 2, 2014.
[6] Kamyab Khajehei, Role of virtualization in cloud computing, International Journal of Advance Research in Computer Science and Management Studies, Volume 2, Issue 4, 2014.
[7] K C Gouda, Anurag Patro, Dines Dwivedi and Nagaraj Bhat., Virtualization Approaches in Cloud Computing, International Journal of Computer Trends and Technology (IJCTT) – volume 12 Issue 4, 161 – 166, 2014.
[8] Kevin Lawton, Bryce Denney, N. David Guarneri, Volker Ruppert, Christophe Bothamy, Edited by Michael Calabrese, Stanislav Shwartsman, available on http://bochs.sourceforge.net/doc/docbook/user/cpu-models.html on December 27, 2016 at 20:17.
[9] Trend Micro, virtualization and cloud computing security best practices, Securing your journey to the Cloud, www.trendmicro.com accessed on December 27, 2016 on 19:32.
[10] EMC, Virtualizing Business critical applications technology concepts and business considerations, White paper, 2010.
[11] VMware, Virtualization overview, White paper, 2006.
[12] David Merrill and Michael Heffernan, Hypervisor Economics: A framework to identify, Measure and reduce the cost of virtual machines, White paper, Hitachi, 2011.
[13] Brandon Baker, Windows Server Virtualization & The Windows Hypervisor, Microsoft, 2007.