Understand the Workload Interference of Cloud Virtualizations

UROP Research Paper

Manjul Sharma

University of Minnesota, Duluth, United States of America
Submitted April 2021

Abstract

Cloud computing has grown in popularity and has become a go to solution for the computing needs of people and businesses. This introduces virtual machines to a variety of increased workloads and understanding their workloads is important inorder to optimize performance and power consumption. In this research we test the performance of Virtual Machines like VMware and KVM by stress testing based on CPU performance, I/O intensive workload and Network stress. Data is individually collected from both these Virtual Machines and analyzed to find patterns and understand the performance of these machines in relation to each other.

1. Introduction

Cloud computing was first introduced by the former Google CEO Eric Schmidt in 2006, who called it the “next big opportunity”. Since then cloud computing services have taken over the world of computing as according to Columbus (2018) around 83% all enterprise workloads will be in cloud [1]. Cloud computing allows the delivery of different services which previously would require one to set up servers and infrastructure, but with cloud computing these services are offered through the internet. Cloud computing has definitely become a major provider of computing to people and businesses, since it provides multiple services leaving it exposed to pressure from different types of workloads. Thus, it is important to understand the effects of these work loads.

This research focuses on understanding the performance of virtual machines (VM) under different types of workloads. Here I am using two popular VM platforms, namely VMware and Kernel-based Virtual Machine (KVM). The reason for choosing these VM platforms is firstly, they offer different levels of accessibilities in terms of cost, VMware charges its users for using their platform where as KVM is a open source tool, and secondly, they are different types of hypervisors, VMware is a type 1 hypervisor and KVM is a mix of both type 1 and type 2 hypervisor. A hypervisor is the software that allows us to create and run VMs, type 1 hypervisors run directly on physical hardware and the operating system (OS) of the users runs on the hypervisor, whereas a type 2 hypervisor runs within the OS that runs on physical hardware.

This paper includes four sections which are Experiment Configuration, Measurement and Observation, Further Discussion and Conclusion. Initially, the paper explains the equipment, procedure and data used to conduct the research. Next, it lists the various results obtained using the mentioned methods. Then, the paper discusses the results obtained and how

effectively I accomplished my objectives. Finally, it concludes the research and offers a reflective statement on my Undergraduate Research Opportunity Program experience.

2. Experiment Configuration

2.1 Devices

The only piece of hardware used in this research was a laptop installed with an Intel Core i7-6700HQ CPU, 12 gigabytes of RAM, a 250 gigabyte SATA SSD and a fresh install of Ubuntu (a linux OS distribution). This hardware was used to install both VMware and KVM on it and it is crucial to ensure that the VMs created on both the VM platforms are identical in characteristics, in order to ensure complete fairness. So, both VMware and KVM have 8 processors, 4 gigabytes of memory and 25 gigabytes of storage. Other softwares utilized to conduct this research were sysbench (Kopytov, 2017) and iPerf (Dugan et al., 2015).

2.2 Procedure

In this experiment, I will test the VM's performance by conducting stress tests based on three different categories which are CPU stress, Input/Output (I/O) stress and finally Network stress. The CPU stress test is conducted by using sysbench which forces the CPU to calculate and verify primes numbers by doing standard division of the number by all the numbers between 2 and the square root of that number. This CPU stress is run while I control the number of threads from 1 to 8, threads are a way to indicate how much work can be done on a CPU at once, a higher number of threads means that more work can be done on the CPU. Next, the I/O stress is tested which is again done using sysbench where a significant amount of test files are created and then read & write operations are performed while I control the number of threads from 1 to 8. Finally, the network stress is performed using iPerf which allows me to set up and send traffic over a client and server, in this case the VMs are separately tested as the client and server while I control the number of threads by performing CPU stress on threads ranging from 1-8.

2.3 Data

There is different data collected for each of the stress tests performed. The data obtained from the CPU stress test is in events performed per second versus the number of threads. Next, the data collected from the I/O stress is in mebibyte per second (MiB/Sec) for each read and write operation versus the number of threads. Finally, the data resulted from the network stress is in KiloBytes per second (KBytes/Sec) from the sender and receiver, where the VMs are tested as the client and server individually versus the number of threads.

3. Measurement and Observation

The results of this research are divided into three parts which are the CPU stress, I/O stress and Network stress results. The results of each part follow their own trends irrespective of the trends in the other results, the complex nature of the results showcases the complexity of VMs and the various factors that affect them.

3.1 CPU stress test

The results of the CPU stress demonstrated a clear linear relationship between the events per second processed by the CPU and the number of threads for both KVM and VMware. When comparing the results between VMware and KVM we can notice that KVM usually has a slight upper hand in terms of events per second but this trend remains constant as the number of threads increases. In figure 1 a line graph is showing this relationship.

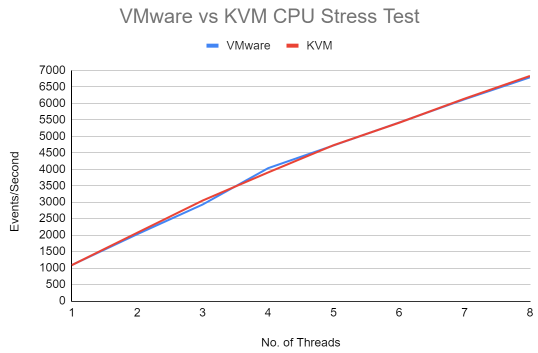

Figure 1: Relationship between the number of threads (x axis) and events per second (y axis). The blue line represents KVM and the red line represents VMware.

3.2 I/O stress test

The results of the I/O stress indicate a direct relationship between read/write capabilities and available threads. However the VMware VM outperformed the KVM VM by around 10 times. In figure 2 a line graph is showing this relationship.

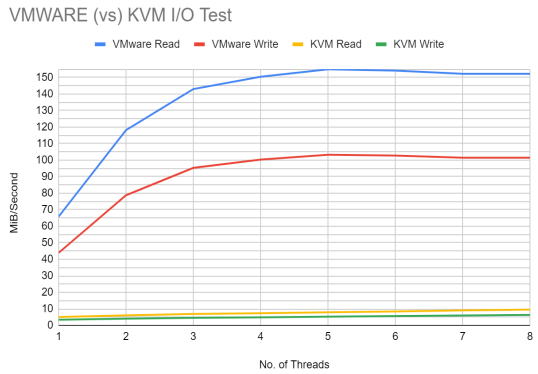

Figure 2: Relationship between the number of threads (x axis) and Mebibytes per second (y axis). The blue and red line represents read and write for VMware and the yellow and green line represents read and write for KVM respectively.

3.3 Network stress test

The results of the network stress test indicate a consistent trend with decreasing bitrate of both VMware and KVM (as client and server) as more threads are put under CPU stress. Here the bitrate of KVM outperforms that of VMware by 10 times. In figure 3 and 4 line graphs are showing this relationship.

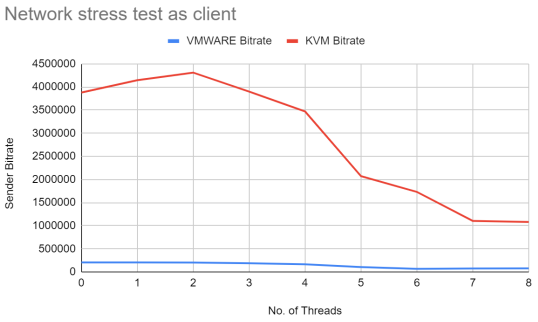

Figure 3: Relationship between the number of threads (x axis) and bitrate (y axis) with the VM as client. The blue line represents VMware and the red line represents KVM.

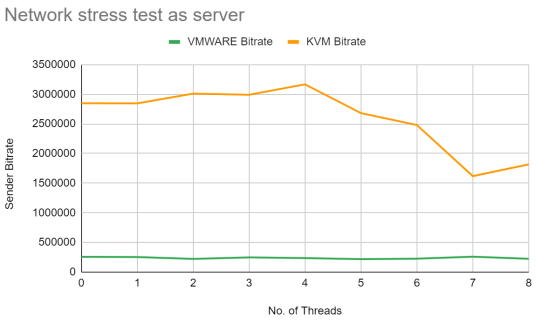

Figure 4: Relationship between the number of threads (x axis) and bitrate (y axis) with the VM as server. The green line represents VMware and the blue line represents KVM.

4. Further Discussions

This research showcases the complexity of VMs and attempts to understand their complexity with three simplified tests. In this research I accomplished my objectives of understanding VM performance which correlates to their power consumption, but I also found various datasets that were contrary to my previous understanding, so this research offered me a great learning opportunity. The CPU stress test showcased the importance of having multiple threads in a VM as they allowed for a linear increase in the events per second processed by the CPU. The lack of I/O performance in KVM compared to VMware showcases its lack of ability to perform high speed read and write operations, this makes the VM made using VMware as the ideal choice for I/O intensive tasks. The network stress test indicates VMware as the best performing VM in terms of bitrate which makes it the ideal choice over KVM for network

intensive workloads. The objectives of this research were accomplished by collecting multiple data on each of the three categories of tests while maintaining a stable environment for these tests to occur. The tests were conducted using trusted technologies like sysbench and iPerf which have widely been used by the computer science community to stress systems. The last objective was to actually understand the data collected to find trends and patterns, this was done using graphs which helped me showcase my findings in a visually understandable format, finally, these trends were recognized and explained in this research paper.

5. Conclusion

To conclude, we can clearly observe that both VMware and KVM offer optimal CPU performance, whereas VMware offers better I/O performance and KVM offers better Network performance. My undergraduate research experience allowed me to stretch beyond what I had known while studying as a computer science student, this enabled me to experiment with my existing knowledge while absorbing new information and techniques. It specifically allowed me to learn more about the field of VMs, my interest in VMs first started during my internship and the UROP gave me the opportunity to further learn and research this field. This experience also gave me a chance to explore the methods and practices involved in professional research that is conducted by professors of the University of Minnesota and allowed me to learn and grow with them. This opportunity taught me the invaluable skill of identifying trends and patterns in data as it is usually just a bunch of numbers but deeper meaning can be extracted from it by careful analysis. One of my biggest takeaways from this research experience is to make our existing technologies better, with proper research, existing technologies can be improved in ways that surpass many new developments. When existing technologies are improved we not only improve their performance but also save time, money and energy. I would also like to thank my research mentor Dr. Haiyang Wang for his continued guidance throughout this research.

Reference

[1] Columbus, L. (2018, January 25). *83% Of Enterprise Workloads Will Be In The Cloud By 2020*. Forbes. https://www.forbes.com/sites/louiscolumbus/2018/01/07/83-of-enterprise-workloads-will-be-in-the-cloud-by-2020/?sh=222fecdb6261

[2] Kopytov, A. (2017, October 26). *sysbench*. GitHub. https://github.com/akopytov/sysbench

[3] Dugan, J., Elliott, S., Mah, B. A., Poskanzer, J., & Prabhu, K. (2015, January 9). *iPerf*. Iperf. https://iperf.fr/