Performance Analysis of Virtual Machine in Cloud Architecture

Dr. Kiran V[1], Akshay Narayan Pai[2], Gautham S[3]

[1]*Associate Professor, Dept. of Electronics and Communication, R.V. College of Engineering, Bangalore, INDIA -560102*

[2,3]*Student, Dept. of Electronics and Communication, R.V. College of Engineering, Bangalore, INDIA -560102*

[1]*kiranv@rvce.edu.in,* [2]*akshaynarayanp.ec17@rvce.edu.in,* [3]*gauthams.ec17@rvce.edu.in*

Abstract: *Cloud computing is a technique for storing and processing data that makes use of a network of remote servers. Cloud computing is gaining popularity due to its vast storage capacity, ease of access, and diverse variety of services.When cloud computing advanced and technologies such as virtual machines appeared, virtualization entered the scene. When customers' computing demands for storage and servers increased, however, virtual machines were unable to match those expectations due to scalability and resource allocation limits. As a consequence, containerization became a reality.Containerization is the process of packaging software code along with all of its essential components, including as frameworks, libraries, and other dependencies, such that they may be separated or separated in their own container.The programme operating in containers may execute reliably in any environment or infrastructure. Containers provide OS-level virtualization, which reduces the computational load on the host machine and enables programmes to run much faster and more reliably.Performance analysis is very important in comparing the throughput of both VM based and Container based design. To analyze it same web application is ran in both the designs. CPU usage and RAM usage in both the designs were compared. Results obtained are tabulated and Proper conclusion has been given.*

Keywords: **Virtualization, Containerization, CPU Consumption, Throughput.**

1. INTRODUCTION

Today, cloud computing is popular as a main computing paradigm in which the customer may obtain the most computer resources on demand. Cloud computing is a powerful computing architecture in which distant server network for data storage and tasks are allocated to a network-accessed mix of connectivity, software and service. Cloud computing has a huge processing capacity, typically in conjunction with server virtualization technologies such as VMware ESX Servers, Citrix Xen servers, and parallel processing using distributed, large-scale computer clusters. This server and communication network is known entirely as the cloud. Additional essential elements in cloud computing are their qualities such as reliability, scalability and resource elasticity, making cloud computing the ideal computing paradigm. Cloud may be used by anybody in the internet to store their data in the cloud and utilise their enormous resources. Depending on customers' demands, all the service and implementation options offered by cloud computing can be employed. Cloud computing plays a key role in providing the physical resources like CPUs, disc storage and broadband network, for example, in virtualization and containerization. The virtualisation of the OS, server, and storage device or network resources is nothing but development of a virtual version of resources. In the Cloud, these physical resources are considered to be a pool of resources, which may thus be used on demand.

Containerization has arisen, on the other hand, as an alternative or complement to virtualization as an important topic in software development. In order to work safely and uniformly on any infrastructure it involves enclosing or packaging the software and all its dependencies. VM, Dockers, containers, and kubernetes are some of the cloud technologies based on virtualization and containerization. Virtual Machine is a virtual machine which runs the OS and application like a physical computer. Containers provide a logical packaging structure from which programmes or applications can be removed from the present environment. The decoupling allows you to rapidly and reliably execute a container-based application, whether it's a private data center, the public cloud or simply a personal laptop developer. The objective of containers and virtual machines is similar, i.e. to divide an application into a self-contained unit that may be functioning anywhere. Unlike a VM that virtualizes hardware, a container offers virtualization at the operating system level by abstracting the "User Space." And VMs execute programmes with multiple OS instances. But in a single OS instance a container accomplishes the same and minimizes the strain generated by a number of OS instances. Docker offers tools and a platform to manage the container life cycle, create, deploy and use containers in the production environment for your application and supporting components.

2. LITERATURE SURVEY

The Paper referenced [1] discusses about Managing Tradeoff between cost and performance during VM deployment. It also gives overview about Optimization Analysis of Performance. And Pricing model in cloud computing including quality of service, workload of application, Profit and so on.Modelling cloud system as an infinite capacity M/M/N queuing model where N represents number of VMs and Queue of the system is equivalent to queuing discipline First Come First Serve (FCFS).

The Paper referenced [2] Gives solution for increased power consumption by cloud data centers. Also discusses Control over number of VM Migrations. Performance analysis based in energy consumption, SLATAH metric, number of hosts shut down and number of migrations.

The Paper referenced [3] focuses about User experience oriented VM placement method. And also discusses about Setting CPU utilization threshold for energy saving. Also, gives brief overview on host management module which manages the hosts of the cloud datacenters in an energy saving way using the comparison ratio defined by number of VMs carried by current host divided by maximum number of VMs that the current host can use.

The Paper referenced [4] discusses about Dynamic Consolidation and Minimization of migration thrashing. Usage of several metrics to compare the efficiency of scaling algorithms. Such as thrashing index metric (T.I), SLA violation metric and Average number of VM migration metric.

The Paper referenced [5] discusses Comparison of Different algorithms for load balancing, Response time and throughput analysis. Performance parameters like response time, data center processing time and throughput were analyzed thoroughly to average out the overall performance improvement analysis.

3. METHODOLOGY& IMPLEMENTATION

Brief outline of the design,

➢ To analyze how Scaling and other parameters affect the VM performance.

➢ To deploy different VMs in Oracle Hypervisor environment on physical machine.

➢ To use Docker Desktop software for creating and maintaining Containerized images in Windows Environment.

➢ To compare the image with Containerized Image and VMs deployed.

VM and Container based design are subjected to performance analysis considering CPU usage and RAM consumption by running a web page which is common to both the designs, detailed analysis of this would be given in this chapter.

In the VM based design, the complete cloud behavior when it is deployed using simply Vertical Scaling approach to handle On-Demand Scaling across Virtual Machines utilizing a local PC (acting as a cloud) and an open-source Hypervisor (Virtual Box) launched by a well-known firm Oracle. Running a web page can be considered for On-demand scaling, while implementing this factors like CPU usage and RAM consumption were observed and analyzed.

The packaging of software code together with all its necessary components such as frameworks, libraries, and the other dependencies such that they are isolated or separated in their own container this served in reducing the CPU usage and RAM consumption in Container based design.
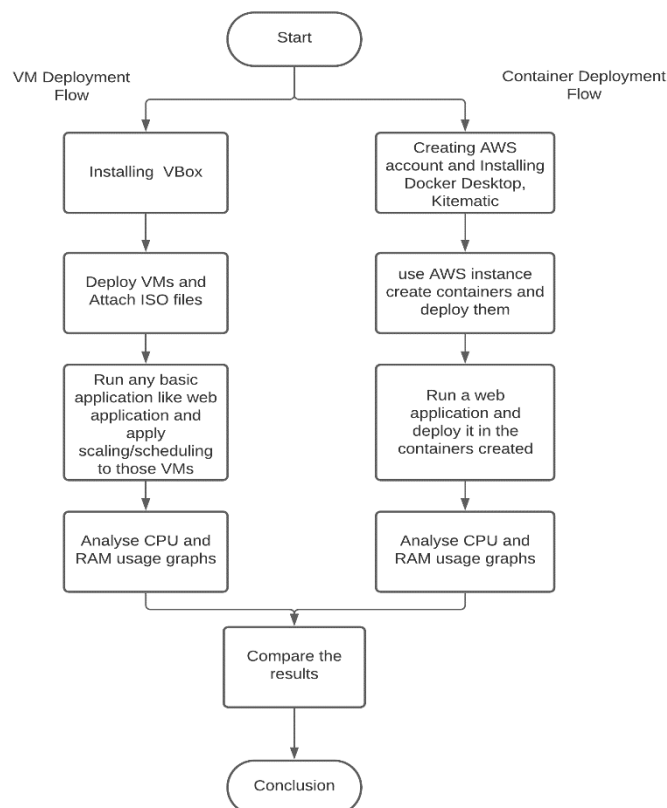

Fig.1 Flowchart of the Project

The application or software within the container can be moved or run consistently in any environment or on any infrastructure. It is independent of the operating system of the environment or infrastructure. The container acts as a sort of bubble or a computing environment which surrounds the application. It also keeps it independent of its surroundings. It is basically a portable and fully functional computing environment.Moving the application to a new environment might be difficult as the code might not then be compatible with the new environment. There would be errors, bugs and glitches which would require fixing. A solution to this problem is containers which are an alternative to coding on one operating system or platform.

Containers allow one to run multiple applications or processes on a single OS instance, making containers quicker due to reduction in workload and are more stable than VMs, which require multiple OS instances and are extremely slow.

The major factors that decide the performance of a VM were determined and analyzed to check if any degradation of performance is there in VM based design and Container based design.

In container based design, AWS cloud instance is used to provide cloud interface and Docker desktop is used as cloud based tool to create containers having their own base images. Docker desktop can be used only in windows based OS, in UNIX based OS this is not needed. HTML code is developed to run a web applications on these created containers.

But running different static pages in individual pods is not feasible as it poses linking problem. So, a legacy Docker related software, Kitematic was used for giving web application a better GUI. This was achieved by changing Persistent volume of container which was earlier deployed and replacing it with static web application's pages.
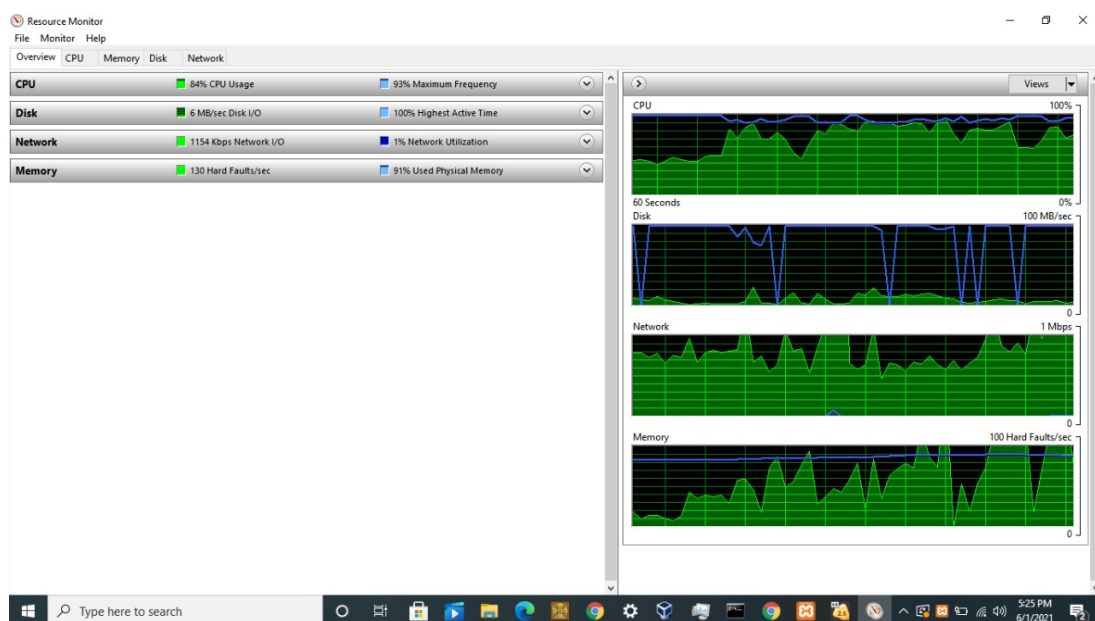

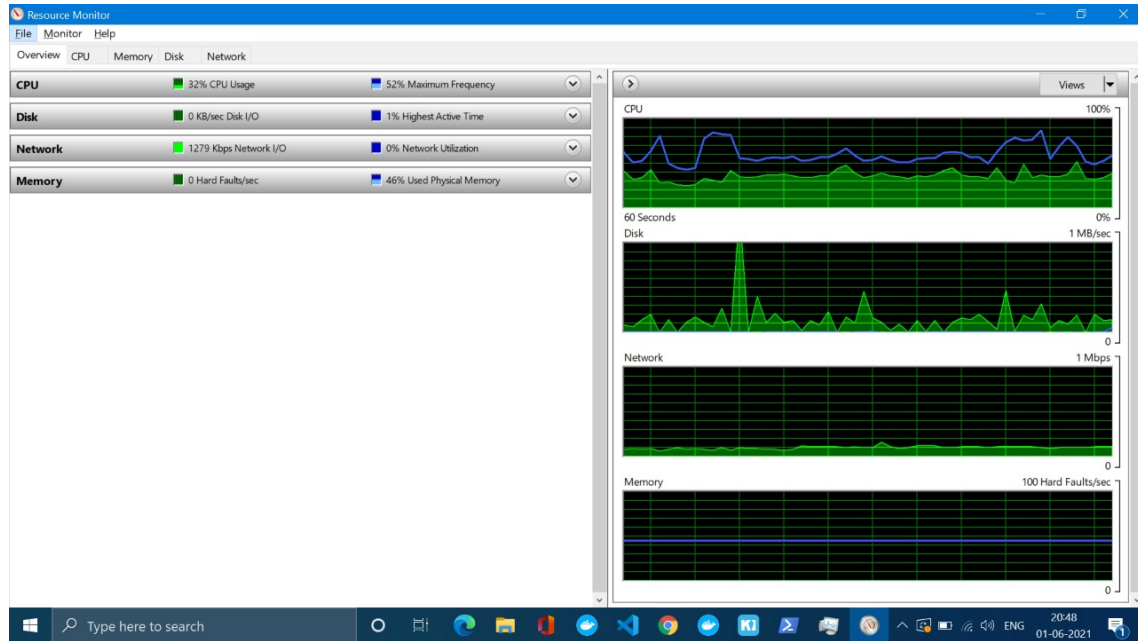Fig.2 Resource monitor graph for VM deployment

Fig.3 Resource monitor graph for Container deployment

The Figs. 2 and 3 shows CPU Usage and RAM consumption of VM Based Design and Container Based Design.

Container based design puts less load on host machine due to its advantage of running the application on cloud.This decreases CPU usage and RAM consumption in host machine leading to increase in its performance.

Parameters	Virtual Machine	Container
CPU Usage	When running an application in VM's Virtualised environment the CPU usage was around 80%	When running an application in Docker's containerized environment the CPU usage was around 30%
RAM Consumption	When above scenario was evaluated for RAM usage, VM consumed around 90%	When above scenario was evaluated for RAM usage, Container consumed around 45%

From the results obtained conclusion can be drawn that container architecture gives better performance and consumes less CPU usage and less RAM consumption than VM architecture, which are really needed for running high-end applications in cloud environment under high load condition. Under light load condition, VM architecture is preferred.

5. CONCLUSION

Cloud service providers meet the needs of their customers by allocating Virtual Machines to them. These Virtual Machines contain resource instances required by the client, such as memory instances, web service instances, computation instances, and so on. All VMs offered by service providers operate on Physical Machines (PMs) that operate on the user's side and on which his application runs.

As previously said, in order to obtain greater throughput out of a cloud-based system, it is critical

to manage the cloud resources and instances appropriately and effectively. Scaling strategies can be used to accomplish this. The cloud is a limitless resource pool, however in the imitated model, a laptop/desktop serves as an alternative to the cloud but has a restricted resource pool in case of VM deployment. This project's findings demonstrate that this little imitated model can handle all edge scenarios that may arise in a real-world setting when using VM based design.

In case of Container based design, AWS cloud instances was used to create containers and pods. Each pod is deployed with one base image which is altered with web application. And performance analysis of this design considering CPU usage and RAM consumption was done and was compared with VM based design, which showed container based design are better in accordance with CPU and RAM consumption than VM based design.

6. FUTURE SCOPE

Nowadays, cloud technology is attempting to shift away from the need on VMs to complete activities, in favor of a more efficient and thriving postulation known as "Containers." A container is a piece of software that combines code and its resources into a single package, allowing programmes to function fast and efficiently at the user's end. One of the benefits of containers is that they allow you to virtualize an operating system so that various workloads may run on a single OS instance, whereas VMs virtualize the hardware to run multiple OS instances.

In addition to all of these in containers, Docker Engine is employed, which eliminates the need for a Hypervisor such as Virtual Box, which is used in the project. The distinction between VM-based cloud resource management and Container-based cloud resource management architectures.

7. REFERENCES

1. B. Wan, J. Dang, Z. Li, H. Gong, F. Zhang, and S. Oh, "Modeling analysis and cost-performance ratio optimization of virtual machine scheduling in cloud computing,"IEEE Transactions on Parallel and Distributed Systems, vol. 31, no. 7, pp. 1518–1532, 2020.
2. J. Singh, J. Sidhu,et al., "Comparative analysis of vm consolidation algorithms forcloud computing,"Procedia Computer Science, vol. 167, pp. 1390–1399, 2020.
3. S. Pang, K. Xu, S. Wang, M. Wang, and S. Wang, "Energy-saving virtual machine placement method for user experience in cloud environment,"MathematicalProblems in Engineering, vol. 2020, 2020.
4. X. Liu, J. Wu, G. Sha, and S. Liu, "Virtual machine consolidation with minimization of migration thrashing for cloud data centers, "Mathematical Problems inEngineering, vol. 2020, 2020.
5. Y. Li, D. Ou, C. Jiang, J. Shen, S. Guo, Y. Liu, and L. Tang, "Virtual machineperformance analysis and prediction," in2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), IEEE, 2020, pp. 1–5