SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 2, 2024, 244-252
DOI: 10.35377/saucis...1432389

RESEARCH ARTICLE

Performance Analysis of Open Source Cloud Computing Architectures

Mehmet Zahid Kuzan[1], Abdullah Sevin[2]

[1] Department of Computer and Information Engineering, Sakarya University, Sakarya, Türkiye
[2] Department of Computer Engineering, Sakarya University, Sakarya, Türkiye

Corresponding author:
Mehmet Zahid Kuzan, Sakarya University, Department of Computer and Information Engineering, Sakarya, Türkiye
mehmet.kuzan@ogr.sakarya.edu.tr

Article History:
Received: 06.02.2024
Accepted: 26.07.2024
Published Online: 26.08.2024

ABSTRACT
Organizations prioritize cloud computing as the primary solution for ensuring secure and, most importantly, uninterrupted delivery of data. In today's landscape, there are various architectural frameworks offering cloud computing infrastructures to choose from. It is crucial to understand how advantageous the performance offered by a chosen architecture is when making a selection. This study aims to comprehend how effective open-source cloud computing architectures are in specific usage scenarios, provide a comparative perspective, and assist organizations in making informed decisions when selecting their IT infrastructures. In line with the study's objectives, the network, disk, and processor performances of OpenStack, OpenNebula, and Apache CloudStack architectures on the same hardware are examined based on data-driven methods. The Iperf3 tool is used for network performance measurement, the dd (Linux) tool for disk performance measurement, and the Sysbench tool for processor performance measurement. After conducting performance measurements with these tools, it was observed that the OpenNebula architecture outperformed the other architectures in terms of overall performance.

Keywords: Open source, Performance analysis, Cloud computing

1. Introduction

Data access difficulties, data loss, or damage can disrupt business continuity in organizations and lead to reputation losses [1]. Cloud computing is considered a priority solution for organizations to ensure the secure and, most importantly, uninterrupted delivery of data [2]. In today's landscape, organizations can prefer various cloud computing architectures[3]. When selecting these architectures, criteria such as cost-effectiveness and customizability can be used in decision-making. Still, it is crucial to know how advantageous the performance offered by the chosen platform is in meeting the needs [4].

This study, titled "Performance Analysis of Open Source Cloud Computing Architectures," aims to understand how effective open-source cloud computing architectures are in specific usage scenarios, provide a comparative perspective, and assist organizations in making informed decisions when selecting their IT infrastructures. In the scope of the performance evaluation conducted in this study, the data-driven methodology proposed by Li et al. for performance evaluation in cloud services will be followed [5]. According to this methodology, to accurately conduct the evaluation, requirements must first be determined. For this, questions defining the purpose should be prepared. Once the purpose is determined, the performance metrics need to be evaluated, and the shaping of the test design process needs to be resolved. After completing the test design process, tests should be conducted based on the defined metrics, and the results obtained from the tests should be analyzed.

The study will focus on open-source cloud computing architectures, specifically OpenStack, Apache CloudStack, and OpenNebula architectures. The aim is to measure these architectures' network, disk, and processor performance. In this context, network bandwidth, disk read speed, disk write speed, and processor speed are designated as performance metrics. After preparing the relevant test environments, performance evaluation tools will be used for metric measurements. The results obtained from the measurements will be used to evaluate the performance of these architectures.

2. Related Works

This section examines previous studies conducted in the literature to acquire knowledge in the research field and interpret research findings. Eren investigated the working logic of virtualization systems, considering performance and usability among applications [6]. The study observed disk, processor, and memory resource changes in virtualization environments. The results showed that increasing the allocated resources improved performance. A study by Doğru aimed to facilitate users' selection between hypervisor and container-based virtualization technologies in cloud computing infrastructure by measuring efficiency [7]. The study observed how disk, memory, and processors performed different tasks, concluding that container virtualization technology outperformed hypervisor virtualization. Elmas conducted a study to observe the efficiency of physical and virtual servers in cloud computing regarding service continuity [8]. The study monitored the memory, processor, and disk performance of randomly selected physical and virtual servers in specific scenarios. The results showed no significant differences between the server types except for memory usage. Ataş studied the processor, disk, and network performance evaluations of Cloud Foundry, Heroku, and Openshift cloud computing platforms [9]. The study involved tests based on four scenarios, revealing that each platform performed differently in various functions. Husain and colleagues conducted a performance analysis of OpenStack and Eucalyptus architectures in 2018 [10]. The study involved performance tests for processors, memory, disk, and network in two hardware configurations. The results indicated that OpenStack's architecture had better network, memory, and processor performance than Eucalyptus. Yadav conducted a study in 2013 examining the features of Eucalyptus, OpenStack, and OpenNebula architectures [11]. The study discussed various features and differences, such as programming languages, virtualization, storage, and image management. Bedi and colleagues conducted a study in 2018 examining the features of OpenNebula, CloudStack, Eucalyptus, and OpenStack architectures [12]. The study covered architectural components, licensing types, development communities, and their positions in the cloud ecosystem. Saisree and Shitrhart conducted a 2022 study examining the features of Eucalyptus, OpenStack, and CloudStack architectures [13]. The study discussed operational features, communities contributing to the development of these architectures, and development models. Mohammed and Kiran conducted a technical evaluation study in 2015, assessing Eucalyptus, CloudStack, OpenStack, OpenNebula, Nimbus, Xen Cloud Platform (XCP), OpenIoT, and AbiCloud architectures and solutions [14]. The study covered the working principles of cloud computing architectures and the features of open-source cloud computing platforms. Işık conducted a performance evaluation on the OpenStack cloud architecture with hypervisor and container virtualization solutions [15]. The study used benchmarking tools to examine processor, memory, network, and disk performance. The results indicated that container virtualization has advantages over hypervisor virtualization in different workloads. Peng and colleagues have examined the technical specifications of the Abicloud, Eucalyptus, Nimbus, and OpenNebula architectures [16]. Observations reveal that each architecture has its unique strengths.

3. Definitions

3.1. Cloud Computing

Although there are many definitions related to cloud computing, the definition generally referred to in the literature is the one made by the National Institute of Standards and Technology (NIST) of the United States. This definition defines cloud computing as a model with configurable resources (computer networks, databases, servers, applications, services, etc.) that can be rapidly provisioned and released with minimal management effort [17].

3.2. Open Source Cloud Computing Architectures

Cloud computing systems can be divided into two parts: the front end, the user interface for accessing the cloud, and the back end, which consists of servers that make up the system [4]. Communication between these two parts is managed by middleware, which follows specific rules and protocols through network connections [4].

Cloud computing architectures are design models to create the cloud computing environment [18]. These architectures determine the components that make up the cloud environment for the services to be provided, how these components will connect, and the infrastructure needs for the services to be offered [18]. Open-source cloud computing architectures are systems created as open-source software and designed to provide cloud computing services [19].

3.3. OpenStack

OpenStack is an open-source cloud computing platform initiated in 2010 under the Apache license by Rackspace and NASA to build, manage, and deploy infrastructure as a service [20], [21]. The platform utilizes virtualization technologies on hardware, allowing users to modularly access the computing, storage, and networking services they need through a web interface [22]. The fundamental architectural components of OpenStack are as follows [20], [21] [22];

- Compute: Manages resources and enables the creation and scaling of virtual machines through the Nova component.
- Storage: Provided through Swift (Object Storage) component offering object storage services and Cinder (Block Storage) component providing block-based storage.
- Networking: Network resources are managed through the component called Neutron. The Neutron component provides network services such as virtual networks, subnets, and routing.
- Orchestration: Automates application deployment with the component named Heat.
- Image Service: Manages disk images of virtual machines using the component called Glance.

- Monitoring: Monitors the system using the Telemetry component, allowing the creation of alarms and notifications.
- Dashboard: Manages cloud resources through a web-based user interface with the component named Horizon.
- Identity Management: Provides identity management and user authorization with the Keystone component.

3.4. Apache CloudStack

Apache CloudStack is an open-source cloud computing platform initiated by Citrix Systems in 2011[23]. It is designed with a modular structure and aims to create and automate virtual machines and other infrastructure components. The key architectural elements of CloudStack include [12]:

- Management Server: The primary component that manages the CloudStack cloud environment. It processes user requests, manages resources, creates virtual machines, stops them, and performs other management tasks.
- Database: Used to store the state of the cloud infrastructure, user information, virtual machine configurations, and other essential data.
- Cloud Infrastructure: CloudStack has an infrastructure layer that provides access to the resources of physical or virtual machines. This layer includes storage, networking, and virtual machines.
- Network: Manages virtual networks and network services in CloudStack. This layer enables users to create, configure, and manage virtual networks and network services.
- Hypervisor: CloudStack supports various virtualization technologies. This layer ensures integration with a specific hypervisor (such as KVM, Xen, VMware, etc.) and manages the operation of virtual machines on this hypervisor.
- Storage: Manages different types of storage in CloudStack. The storage layer oversees storage pools and stores disk images of virtual machines.
- API (Application Programming Interface): This layer is the interface through which CloudStack interacts with the outside world. It hosts APIs used to access cloud resources programmatically.
- User Interface: The User Interface is a web-based interface where users can visually manage cloud resources. Users can use this interface to manage virtual machines, configure networks, and monitor other cloud resources.

3.5. OpenNebula

OpenNebula was initiated by Ignacio M. Llorente and Ruben S. Montero in 2005 [11]. It is an open-source cloud computing platform and virtualization management software. The core architectural components of OpenNebula include [11]

- Core: This component handles user requests for resource allocation and communicates with other components to manage the cloud infrastructure. The OpenNebula Core manages the cloud infrastructure and controls fundamental cloud functions.
- Compute: Responsible for managing virtual machines in the cloud environment. It facilitates the creation, startup, shutdown, resizing, and disabling of virtual machines.
- Storage: Manages the storage resources of virtual machines. Transfer Manager (TM) and Datastore Manager (DS) are used to create, resize, copy, delete virtual disks, and manage file systems.
- Network: Manages tasks related to network components. This component handles the management of virtual networks, routers, and other resources.
- SQL Database: A SQL database stores system configuration information and user data.
- User Interface (Front-end): This web-based component allows users to manage and monitor cloud resources. The user interface is provided by a component called OpenNebula Sunstone.
- Tools Layer: This component tracks virtual machine statuses, resource usage, and user activities.
- Marketplace: A component where users can download ready-made virtual machine templates and applications.

4. Material and Method

Inspired by existing studies in experimental computer science and principles of experimental design, Li and colleagues developed the DoKnowMe methodology for the performance evaluation of software and computing systems [5]. The evaluation process of DoKnowMe consists of a sequential main process and recurring experimental activities. Each evaluation step uses inputs, relevant evaluation strategies, and outputs. During this process, evaluation concepts are defined, performance metrics are determined, and templates are created for repeatability. According to this methodology, requirements must be determined to conduct the evaluation accurately. For this purpose, questions defining the objectives should be prepared. Subsequently, determining the performance metrics to be evaluated and shaping the test design process is essential. It is crucial to define why the tests for performance evaluation are conducted to outline the framework of the study. In this study, the following questions have been prepared to evaluate the network, disk, and processor performances of OpenStack, Apache CloudStack, and OpenNebula architectures:

- Question 1: Among OpenStack, Apache CloudStack, and OpenNebula architectures, which one exhibits higher network performance?
- Question 2: Among OpenStack, Apache CloudStack, and OpenNebula architectures, which one demonstrates higher disk performance?
- Question 3: Among OpenStack, Apache CloudStack, and OpenNebula architectures, which one shows higher processor performance?

Network performance will be assessed based on network speed and bandwidth, disk performance will be evaluated through reading and writing speed from/to the disk, and processor performance will be measured by the calculation speed of the processor in arithmetic operations. The identified performance metrics will be tested using performance measurement tools. iPerf3 will be used to observe network bandwidth. iPerf3 is a widely used open-source testing tool that can generate TCP and UDP data streams and measure the network's efficiency [24]. To measure disk read and write performance, the dd tool (Linux) will be used. The dd (Linux) tool, through specific commands, writes the desired-sized data to the disk in blocks, allowing the measurement of how quickly the data is written to the disk and how quickly the specified-sized data can be read from the disk [25]. The Sysbench tool will be used to measure processor performance. The Sysbench tool enables the calculation of prime numbers up to a user-defined size. It provides metrics such as the time taken during the process and the number of successful operations performed [26]. Table 1 presents the performance testing tools and measurement metrics used in the study.

Table 1. Performance Testing Tools and Measurement Metrics

Source	**Test Tool**	**Measurement Metric**	**Unit**
Network	Iperf3	Bandwidth	MB/s
Disk	dd (Linux)	Writing speed of 256 MB data to disk	MB/s
Disk	dd (Linux)	Writing speed of 512 MB data to disk	MB/s
Disk	dd (Linux)	Writing speed of 1 GB data to disk	MB/s
Disk	dd (Linux)	Reading speed of 256 MB data from disk	MB/s
Disk	dd (Linux)	Reading speed of 512 MB data from disk	MB/s
Disk	dd (Linux)	Reading speed of 1 GB data from disk	MB/s
Processor	Sysbench	Finding prime numbers up to 5.000	s
Processor	Sysbench	Finding prime numbers up to 10.000	s
Processor	Sysbench	Finding prime numbers up to 20.000	s

The minimum hardware requirements specified by the developers of OpenStack, Apache CloudStack, and OpenNebula architectures will be used for testing. In this context, 16 GB of RAM, 120 GB of hard disk, and 8 CPUs are sufficient for each architecture. The primary purpose of preferring hardware configurations with identical specifications is to observe the performance of each architecture on hardware with equivalent features. The hardware specifications to be used for the tests are indicated in Table 2.

Table 2. Hardware Specifications to be Used in the Scope of the Test

Requirements	**Feature**
Processor	Intel® Core™ i7-1255U 12.Gen. 3.50 GHz 10 Core 12 Thread
Memory	Samsung M471A2K43EB1-CWE Ram 3200MHz 32GB
Disk	Samsung MZVLQ512HBLU-00BH1 512GB SSD
Network Interface Card	Realtek RTL8111 10/100/1000M
Operating System	Ubuntu 22.04 LTS (Wayland)

The hardware used in the test has an installed Ubuntu 20.04.6 LTS (Focal Fossa) operating system. Subsequently, the KVM virtualization software was installed on the operating system to provide hardware configuration with identical features for the architectures. After preparing the host machine where the test environments will be set up, the relevant architectures were installed. Each architecture, with hardware specifications of 16 GB RAM, 120 GB disk, and 8 CPUs, was installed on the Ubuntu 20.04.6 LTS (Focal Fossa) virtual operating system as a single node according to the installation guide specified on its official website. After the architectures were installed, the Ubuntu Server 18.04 LTS (Bionic Beaver) operating system was installed as a virtual machine on each architecture to perform network, disk, and processor tests. Access to the virtual machines was established using the SSH protocol, and performance test tools were installed. Network, disk, and processor performance tests for the OpenStack, Apache CloudStack, and OpenNebula architectures will follow the flow outlined in Figure 1.

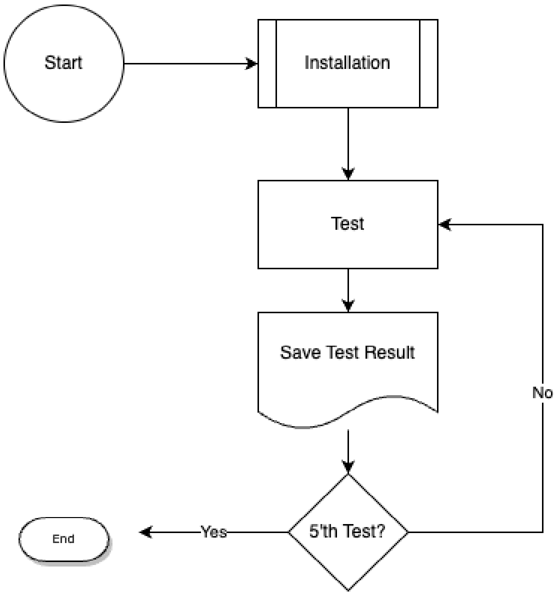

Figure 1. Test Flow Chart

Test Flow:

- Installation: Set up OpenStack, Apache CloudStack, and OpenNebula architectures separately.
- Test: Utilize performance measurement tools to test the identified performance metrics for each architecture.
- Recording: Record the results of each test for further analysis.
- Save Test Result: Repeat each test five times to ensure consistency and reliability of results.

This test plan ensures a comprehensive and reliable evaluation of the network, disk, and processor performances of OpenStack, Apache CloudStack, and OpenNebula architectures. The repetition and averaging approach aims to minimize potential outliers and accurately represent the systems' performance.

4.1. Network Performance Measurement

The study will focus on network speed and bandwidth metrics for network performance. In this context, measuring network speed involves quantifying the data transferred between the client and server in a specific period and observing the achieved bandwidth during this process. In this study, network performance tests will be conducted using the iPerf3 tool over the TCP protocol. The test results have been visualized in Figure 2.

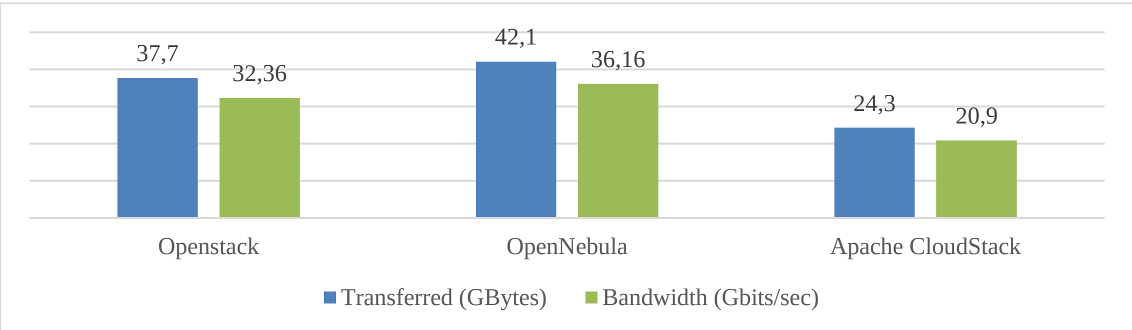

Figure 2. Network Performance Test Results

Based on the average results of the tests conducted with the iPerf3 tool, OpenNebula achieved the highest test score regarding network connection speed and bandwidth, with a data transfer rate of 42.10 GB and a bandwidth of 36.16 GB/s. Then, OpenStack ranked second with a data transfer rate of 37.7 GB and a bandwidth of 32.36 GB/s, while CloudStack came last with a data transfer rate of 24.3 GB and a bandwidth of 20.9 GB/s. It's worth noting that the tests were conducted between two operating systems running on the same Ethernet card, which may have contributed to the higher values obtained. Introducing different devices and physical connections could potentially result in reduced bandwidth. The Apache CloudStack architecture utilizes the EC-2 classic network management structure, which is no longer used by Amazon. Therefore, it can be stated that it lags behind other architectures. The OpenStack architecture manages network services through the Neutron component via APIs. The success of the OpenNebula architecture can be attributed to its superior management of message queues through APIs compared to other architectures.

4.2. Disk Performance Measurement

The study focuses on disk performance measurement using metrics such as disk read and write speeds. In this context, it is essential to observe the time it takes to write certain-sized data to the disk and read the same-sized data from it. Disk read and write performance tests will be conducted using the dd (Linux) tool. The disk reading test results are visualized in Figure 3, and the writing test results are in Figure 4.

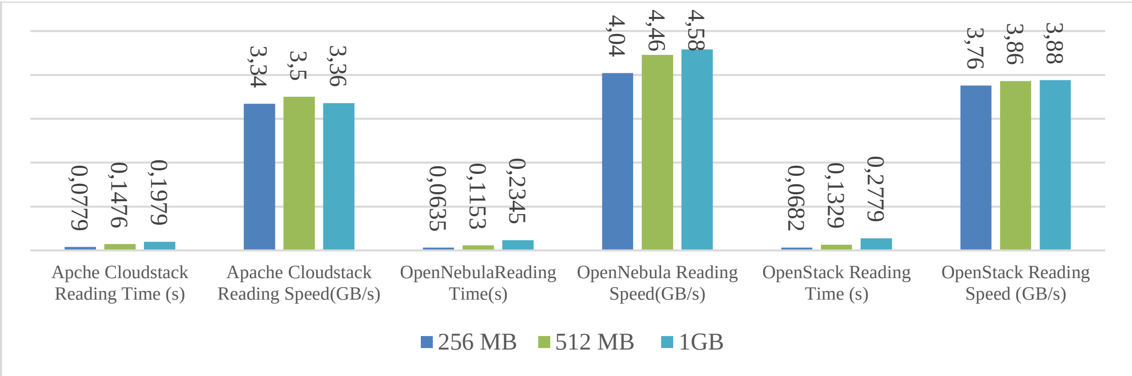

Figure 3. Disk Reading Test Results

Considering that the architecture that performs the most data reading in the least amount of time is considered more successful according to this criterion:

- In the test of reading 256 MB of data from the disk, OpenNebula architecture showed the best performance. Following in performance were the OpenStack architecture and, lastly, the CloudStack architecture.
- In the test of reading 512 MB of data from the disk, OpenNebula architecture showed the best performance. Following in performance were the OpenStack architecture and, lastly, the CloudStack architecture.
- In the test of reading 1 GB of data from the disk, OpenNebula architecture showed the best performance. Following in performance were the OpenStack architecture and, lastly, the CloudStack architecture.

According to the results obtained from the tests, it is observed that OpenNebula architecture is the most successful in disk reading tests. Following the ranking, OpenStack architecture is in the second position, and CloudStack architecture is in the last position. The performance of reading data from disk depends on the way the architecture holds and manages data through its storage components. The Apache CloudStack architecture utilizes primary and secondary storage areas for storage, managed through CloudStack APIs. The OpenStack architecture provides block storage with the Cinder component. OpenNebula architecture, on the other hand, performs storage through data stores named DataStore, and access to these is achieved through a component called Transfer Manager. The fundamental reason for the success of the OpenNebula architecture compared to other architectures lies in the stronger performance of its storage components in reading data from disk.

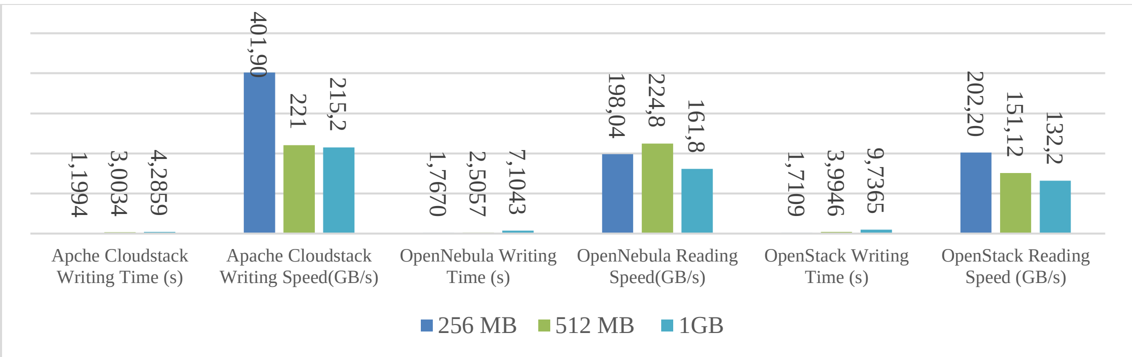

Figure 4. Disk Writing Test Results

Considering that the architecture that can write the most data to the disk in the least amount of time is considered more successful according to this criterion:

- In the test of writing 256 MB of data to the disk, Apache CloudStack architecture showed the best performance. Following in performance were the OpenStack architecture and, lastly, the OpenNebula architecture.
- In the test of writing 512 MB of data to the disk, OpenNebula architecture showed the best performance. The Apache CloudStack architecture followed in performance and, lastly, the OpenStack architecture.
- In the test of writing 1 GB of data to the disk, Apache CloudStack architecture showed the best performance. Following in performance were the OpenNebula architecture and, lastly, the OpenStack architecture.

According to the results obtained, it was observed that the most successful architecture in disk writing tests was Apache CloudStack. In the ranking, OpenNebula architecture followed, and finally, OpenStack architecture was in the last place. The performance of writing data to disk depends on the way the architecture holds and manages data through its storage components. The Apache CloudStack architecture utilizes primary and secondary storage areas for storage, managed through CloudStack APIs. The OpenStack architecture provides block storage with the Cinder component. OpenNebula architecture, on the other hand, performs storage through data stores named DataStore, and access to these is achieved through a component called Transfer Manager. The fundamental reason for the success of the Apache CloudStack architecture compared to other architectures lies in the stronger performance of its storage components in writing data to disk.

4.3. Processor Performance Measurement

The measurement of processor performance in the study will be based on the metric of the processor's computational speed in arithmetic operations. In this regard, observing how quickly the processor completes an arithmetic operation is necessary. Arithmetic operations will be performed using the Sysbench tool to conduct processor performance tests. The Processor Performance Tests' results are visualized in Figure 5.

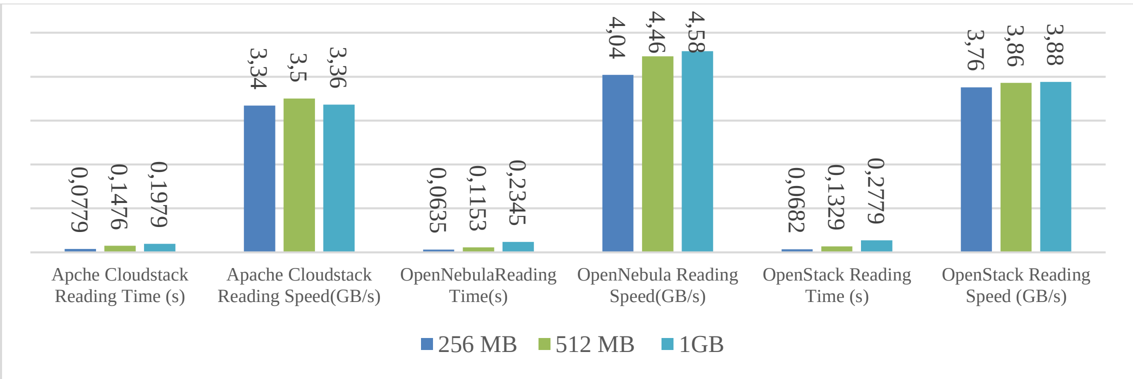

Figure 5. Processor Performance Test Results

Evaluation based on the criterion that the architecture performing the most successful operations in the least amount of time is considered more successful;

- In the prime number finding test up to 5.000, OpenNebula architecture has shown the best performance. Following this, there is the OpenStack architecture, and lastly, the CloudStack architecture.
- In the prime number finding test up to 10.000, OpenNebula architecture has shown the best performance. Following this, there is the OpenStack architecture, and lastly, the CloudStack architecture.
- In the prime number finding test up to 20.000, OpenNebula architecture has shown the best performance. Following this, there is the OpenStack architecture, and lastly, the CloudStack architecture.

According to the results obtained from the tests, it has been observed that the architectures exhibit similar values in the conducted tests. When performance ranking is considered, it can be said that Apache CloudStack architecture outperformed the others in all processor tests. Processor performance in architecture varies based on how the computing components utilize and manage the processor cores. In the OpenStack architecture, the component used for computing tasks is called Nova. This component utilizes a set of tools for creating virtual machines and resource allocation for computing tasks coming from users. In the OpenNebula architecture, the component used for computing tasks is called OpenNebula Daemon. This component utilizes a set of virtualization drivers for creating virtual machines and resource allocation for computing tasks coming from users. In the Apache CloudStack architecture, computing tasks are handled through the Management Server component and the Management Agent component. These components manage the resources of virtual machines, ensuring the fulfillment of user requests. The fundamental reason for the success of the Apache CloudStack architecture compared to other architectures lies in the stronger performance of its computing components in utilizing processor cores.

5. Conclusion and Recommendations

Critical metrics such as processor speed, network speed, and disk read/write speed were determined for performance analysis, and the test preparations were made based on these objectives. While preparing the test environments, it was ensured that each architecture operated on the same hardware configuration. The primary purpose of doing this is to observe the performances of architectures under the same hardware scale. After the test environments were prepared, a virtual server was created for each architecture, and according to the test plan, each performance metric was observed using test tools. Based on the results obtained from the tests, scores were assigned to the architectures for each performance metric. A scoring system was applied where the most successful architecture received 3 points, the second-ranked architecture received 2 points, and the last-ranked architecture received 1 point. Table 1 shows the scores each architecture received after all the tests.

Table 1. Initial Results

Architecture	Disk Writing Test			Disk Reading Test			Network Test	Processor Speed			Total
	256 MB	512 MB	1 GB	256 MB	512 MB	1 GB	Bandwidth	5.000	10.000	20.000	
OpenStack	2	1	1	2	2	2	2	1	1	1	15
OpenNebula	1	3	2	3	3	3	3	2	2	2	24
Apache CloudStack	3	2	3	1	1	1	1	3	3	3	21

According to the scoring table, it is understood that the performance values of the architectures may vary depending on the tested metrics. Based on the results of the disk writing tests, it was observed that the Apache CloudStack architecture is the fastest for writing 256 MB-sized data to the disk, OpenNebula architecture is the fastest for writing 512 MB-sized data to the disk, and Apache CloudStack architecture is the fastest for writing a 1 GB-sized data to the disk. In the results of the disk reading tests, it was observed that the OpenNebula architecture is the fastest for reading 256 MB-sized data from the disk, OpenNebula architecture is the fastest for reading 512 MB-sized data from the disk, and OpenNebula architecture is the fastest for reading a 1 GB-sized data from the disk. The results of the network bandwidth tests showed that the OpenNebula architecture provides the widest bandwidth. Then, OpenStack architecture showed the best performance and Apache CloudStack architecture showed the least performance. According to the results of the processor speed tests for finding prime numbers up to 5.000, 10.000, and 20.000, Apache CloudStack architecture ranked first, OpenNebula architecture ranked second, and OpenStack architecture ranked third. After the tests, the overall scoring resulted in OpenNebula architecture being the first with 24 points, Apache CloudStack architecture being the second with 21 points, and OpenStack architecture being the third with 15 points. It was observed that OpenNebula architecture stands out in terms of network and disk data reading. In this study, each architecture was installed on a single node due to the hardware constraint. Observations indicate that the performance of architectures varies based on how they store, manage, and process data. This variation is closely associated with the success exhibited by the components responsible for network management, computation, and storage within the architectures. The performances obtained by distributing and running the relevant architectures on multiple virtual servers have not been observed. In future studies evaluating the performances of cloud computing architectures, distributing installations to multiple virtual servers according to architectural requirements will contribute to addressing the deficiencies in the literature.

References

[1] İ. Günebakan, "KOBİ'ler İçin Bulut Bilişimin Avantaj ve Dezavantajları," *International Journal of Academic Value Studies*, vol. 2, no. 3, pp. 116–132, May 2016.

[2] C. Zou, H. Deng, and Q. Qiu, "Design and implementation of hybrid cloud computing architecture based on cloud bus," in *Proceedings - IEEE 9th International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2013*, 2013, pp. 289–293. doi: 10.1109/MSN.2013.72.

[3] P. T. Endo, G. E. Gonçalves, J. Kelner, and D. Sadok, "A Survey on Open-source Cloud Computing Solutions," in *VIII Workshop em Clouds, Grids e Aplicações*, 2010, pp. 3–16.

[4] J. W. Rittinghouse and J. F. Ransome, "Cloud Computing: History and Evolution," in *Encyclopedia of Information Systems and Technology - Two Volume Set*, vol. 1, CRC Press, 2011, pp. 178–192.

[5] Z. Li, L. O'Brien, and M. Kihl, "DoKnowMe: Towards a Domain Knowledge-driven Methodology for Performance Evaluation," Aug. 2017, doi: 10.1145/2897356.2897360.

[6] S. Eren, "VMWare ve Hyper-V Sanallaştırma Sistemlerinin Performans ve Yüksek Kullanılabilirliğinin Deneysel Olarak Karşılaştırılması," Yüksek Lisans Tezi, Maltepe Üniversitesi, 2020.

[7] A. Doğru, "Sunucu Sanallaştırma ve Uygulama Sanallaştırma Teknolojileri Performans Karşılaştırması," Yüksek Lisans Tezi, Maltepe Üniversitesi, 2019.

[8] H. Elmas, "Bulut Teknolojisinin Uygulama Sunucularının Yönetimi ve Performansı Üzerindeki Etkisi," Yüksek Lisans Tezi, İstanbul Üniversitesi, 2013.

[9] G. Ataş, "Performance Evaluations of Cloud Computing Platforms," Yüksek Lisans Tezi, Bahçeşehir University, 2013.

[10] A. Husain, M. H. Zaki, and S. Islam, "Performance Evaluation of Private Clouds: OpenStack vs Eucalyptus." [Online].

Available: http://www.publishingindia.com/ijdcc

[11] S. Yadav, "Comparative Study on Open Source Software for Cloud Computing Platform: Eucalyptus, OpenStack and OpenNebula." [Online]. Available: https://www.researchgate.net/publication/262335770

[12] P. Bedi, B. Deep, P. Kumar, and P. Sarna, "Comparative Study of OpenNebula, CloudStack, Eucalyptus and OpenStack," *International Journal of Distributed and Cloud Computing*, vol. 6, no. 1, pp. 37–42, 2018, [Online]. Available: http://www.publishingindia.com/ijdcc

[13] Saisree S. and Shitharth S., *Artificial Intelligence and Data Science*, vol. 1673. In Communications in Computer and Information Science, vol. 1673. Cham: Springer Nature Switzerland, 2022. doi: 10.1007/978-3-031-21385-4.

[14] B. Mohammed and M. Kiran, "Analysis of Cloud Test Beds Using OpenSource Solutions," in *Proceedings - 2015 International Conference on Future Internet of Things and Cloud, FiCloud 2015 and 2015 International Conference on Open and Big Data, OBD 2015*, Institute of Electrical and Electronics Engineers Inc., Oct. 2015, pp. 195–203. doi: 10.1109/FiCloud.2015.106.

[15] G. Işık, U. Gürel, and A. G. Yavuz, "Bulut Ortamlarında Hipervizör ve Konteyner Tipi Sanallaştırmanın Farklı Özellikte İş Yüklerinin Performansına Etkisinin Değerlendirilmesi," *Uludağ University Journal of The Faculty of Engineering*, pp. 981–1002, Aug. 2020, doi: 10.17482/uumfd.605560.

[16] J. Peng, X. Zhang, Z. Lei, B. Zhang, W. Zhang, and Q. Li, "Comparison of several cloud computing platforms," *2nd International Symposium on Information Science and Engineering, ISISE 2009*, pp. 23–27, 2009, doi: 10.1109/ISISE.2009.94.

[17] E. Simmon, "Evaluation of Cloud Computing Services Based on NIST SP 800-145." Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD, 2018. doi: https://doi.org/10.6028/NIST.SP.500-322.

[18] J. Yang, L. Zhang, and X. A. Wang, "On Cloud Computing Middleware Architecture," in *Proceedings - 2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, 3PGCIC 2015*, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 832–835. doi: 10.1109/3PGCIC.2015.46.

[19] T. Cordeiro *et al.*, "Open source cloud computing platforms," in *Proceedings - 9th International Conference on Grid and Cloud Computing, GCC 2010*, 2010, pp. 366–371. doi: 10.1109/GCC.2010.77.

[20] O. Sefraoui, M. Aissaoui, and M. Eleuldj, "OpenStack: Toward an Open-source Solution for Cloud Computing," *Int J Comput Appl*, vol. 55, no. 3, pp. 38–42, Oct. 2012, doi: 10.5120/8738-2991.

[21] T. Rosado and J. Bernardino, "An overview of OpenStack architecture," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, 2014, pp. 366–367. doi: 10.1145/2628194.2628195.

[22] Kevin. Jackson, *OpenStack Cloud Computing Cookbook.* Packt Publishing, 2013.

[23] K. Ahokas, "Comparison of cloud management platforms," 2014.

[24] G. Işık, "Bulut Ortamlarında Hipervizör ve Konteyner Tipi Sanallaştırmanın Farklı Özellikte İş Yüklerinin Performansına Etkisinin Değerlendirilmesi," Yüksek Lisans TEzi, Eskişehir Osmangazi Üniversitesi, 2021.

[25] F. Gomez-Folgar, A. Garcia-Loureiro, T. F. Pena, and R. Valin, "Performance of the CloudStack KVM pod primary storage under NFS version 3," in *Proceedings of the 2012 10th IEEE International Symposium on Parallel and Distributed Processing with Applications, ISPA 2012*, 2012, pp. 845–846. doi: 10.1109/ISPA.2012.128.

[26] A. P. Kurniawan, M. N. Ashar, and F. Hidayat, "Performance Evaluation for Deploying Dockerized Web Application on AWS, GCP, and Azure," in *2023 IEEE International Conference on Control, Electronics and Computer Technology (ICCECT)*, 2023, pp. 346–350.

Author(s) Contributions

The article is multi-author articles. Author 1 conceived the presented idea and performed the calculations. Author 2 developed the theory of this study and supervised its findings. All authors discussed the results and contributed to the final paper.

Conflict of Interest Notice

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical Approval and Informed Consent

It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of Data and Material

Not applicable

Plagiarism Statement

This article has been scanned by iThenticate ™.