Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

June 2026. Vol. 5. No. 3; e-ISSN: 2808-4519

Comparative Performance Analysis of Apache2 and Nginx Web Servers on Ubuntu Server Using the Apache Benchmark Method

Karolus Doweng Koten[1]*, Richard Agung Orlando Berutu[2], Lotar Mateus Sinaga[3], Sem Amuru[4]

[1,2,3,4] *Faculty of Computer Science, Santo Thomas Catholic University*
karolusdowengkoten509@gmail.com[1*], richardberutu5@gmail.com[2], lotarmateus88@gmail.com[3], semamuru93@gmail.com[4]

Abstract

Advancements in information technology have led to an increasing demand for fast, stable, and efficient website services. Web server performance is a crucial factor in maintaining service quality, particularly when handling simultaneous user requests. Apache2 and Nginx are two widely used web servers on Linux operating systems; they possess distinct architectural characteristics that result in differing performance levels. This study aims to analyze and compare the performance of Apache2 and Nginx web servers on Ubuntu Server using the Apache Benchmark method. The research employed an experimental approach involving the installation, configuration, and testing of both web servers within a VirtualBox virtualization environment. The performance metrics evaluated included requests per second, time per request, failed requests, and transfer rate. Testing was conducted across three scenarios featuring varying numbers of requests and concurrent users. The results indicate that both web servers successfully handled all requests without any failures. However, Nginx demonstrated superior performance compared to Apache2, achieving higher requests per second and transfer rates, as well as lower time per request. This study provides comparative performance data for Apache2 and Nginx on Ubuntu Server, serving as a reference for selecting the appropriate web server based on specific implementation requirements.

Keywords: *Apache2, Nginx, Ubuntu Server, Apache Benchmark, Web Server.*

1. Introduction

Advances in information technology have increased the demand for website services across various sectors, such as education, government, business, and public services. This situation emphasizes the importance of having a server system capable of providing fast, stable, and efficient services. A web server is a key component of website service infrastructure, responsible for receiving, managing, and processing user requests through the HTTP or HTTPS protocols.

As the number of internet users and website services continues to grow, the access load handled by servers has also increased significantly. This condition requires the use of web servers that can provide fast, stable, and efficient services, especially when handling multiple requests simultaneously. If a web server is unable to manage a high access load, it may experience increased response time, degraded service performance, or even fail to process user requests.

Web server performance is an important factor in maintaining the quality of website services. Research [1] indicates that a well-performing web server can improve access speed and maintain service stability while handling numerous user requests simultaneously. Apache and Nginx are two widely used web servers for website services. Apache is known for its flexibility in configuration settings and the availability of comprehensive modules. Meanwhile, Nginx is recognized for its ability to handle a large number of user connections. These differences in characteristics cause the performance of the two web servers to vary under certain conditions.

Research [2] shows that web server performance is determined by the server configuration method and the server's ability to handle multiple simultaneous requests. Furthermore, research [3] states that Apache and Nginx exhibit different performance characteristics on the Ubuntu Linux operating system. Other studies also indicate that the choice of web server can affect response speed and website service stability based on server performance testing results [4] [6].

Ubuntu Server is a popular server platform due to its high stability, open-source nature, and compatibility with various server services. The use of Ubuntu Server for web server implementation enables optimal testing according to server management requirements. Based on this

background, this study aims to analyze and compare the performance of Apache and Nginx web servers on the Ubuntu Server operating system based on response speed parameters. The results of this study are expected to determine which web server provides the best performance.

A web server is a service used to manage and provide website access through the HTTP or HTTPS protocols. In this study, Apache and Nginx were selected because both are open-source software and are widely used on Linux operating systems. Apache offers flexibility in configuration settings and a comprehensive set of modules. Meanwhile, Nginx performs well in handling a large number of user connections. Several studies have shown that Apache and Nginx can provide stable performance in managing website services based on Ubuntu and Debian Linux.

Based on previous studies, most research has only compared the performance of Apache and Nginx in specific environments or using different testing parameters. In addition, there are still few studies that specifically evaluate the performance of Apache2 and Nginx on Ubuntu Server using the Apache Benchmark method with the parameters of requests per second, time per request, failed requests, and transfer rate simultaneously. Therefore, this study was conducted to provide an overview of the performance differences between the two web servers in the Ubuntu Server environment using the same testing scenario, so that the results can be used as a reference in determining the most suitable web server for website service requirements [16]. This study is expected to provide information on how Apache2 and Nginx perform on Ubuntu Server, thereby serving as a reference for selecting the most appropriate web server for website development needs.

Several common methods used to evaluate web server performance include Apache Benchmark, Httperf, Apache JMeter, and WRK. Each method has its own characteristics in applying workloads to the server. In this study, Apache Benchmark was selected because of its ability to provide information regarding requests per second, average request time, failed requests, and data transfer rate in an easy and effective manner. This tool is also widely used in research to analyze web server performance.

Ubuntu Server is a Linux-based operating system widely used for server applications due to its high stability, strong security, and support for various network services, such as web servers and cloud computing. Ubuntu is also compatible with Apache, Nginx, MySQL, and PHP, making it a popular choice for website management. Furthermore, this study uses Apache Benchmark to evaluate web server performance based on request parameters, user connections, response speed (response time), and execution time. The purpose of this testing is to determine the server's capability in handling simultaneous access [15], [17].

2. Research Methodology

2.1. Research Method

This study employs an experimental method to analyze the performance comparison between the Apache2 and Nginx web servers on the Ubuntu Server operating system. The experimental method was chosen because the study was conducted through direct performance testing of both web servers based on predetermined parameters. According to [5], the experimental method can be used to determine a server's capability in handling client requests through a benchmarking process.

In this study, both web servers were tested using Apache Benchmark with the parameters of requests per second, time per request, failed requests, and transfer rate. These parameters were used to determine the server's capability in handling simultaneous client access loads [10].

The experimental method in this study consists of several stages, namely reviewing existing literature, preparing the virtualization environment, installing and configuring the web servers, conducting performance testing, collecting the obtained data, and analyzing the test results. These stages were carried out systematically to ensure that the test results clearly and fairly represent the performance of each web server.

2.2. Apache Benchmark

Apache Benchmark is an HTTP-based performance testing tool used to measure the capability of a web server in handling multiple client requests simultaneously. Apache Benchmark, or **ab**, works by sending a number of requests to the server and then generating information regarding server performance, such as requests per second, response time, failed requests, and transfer rate [4].

The use of Apache Benchmark in this study aims to obtain performance data for Apache2 and Nginx on Ubuntu Server through several variations in the number of requests and concurrent users. According to [9], Apache Benchmark is an effective tool for testing web server performance because it provides fast and consistent measurement results [8], [7].

2.3. Research Stages

The research stages carried out in this study consist of several processes, namely literature review, preparation of the testing environment, Ubuntu Server installation, Apache2 and Nginx installation and configuration, testing using Apache Benchmark, and analysis of the test results.

The first stage is a literature review conducted by collecting various references related to the Apache2 web server, Nginx, Ubuntu Server, and the Apache Benchmark method. The second stage is the preparation of the testing environment using Oracle VM VirtualBox as the server virtualization platform.The next stage involves installing Ubuntu Server as the primary operating system. After the operating system has been successfully installed and executed, Apache2 and Nginx are installed and configured separately. Performance testing is then carried out using Apache Benchmark with several variations in the number of requests and concurrent users.

The test results are recorded based on the parameters of requests per second, time per request, failed requests, and transfer rate. Furthermore, the collected data are analyzed to determine the performance differences between Apache2 and Nginx on Ubuntu Server [21].The research stages carried out in this study are presented in Figure 1. These stages illustrate the research workflow, starting from the literature review, preparation of the testing environment, web server installation, benchmarking using Apache Benchmark, and ending with the analysis of the test results.

In general, the research stages were conducted sequentially, starting from the planning process to the analysis of the test results. Each stage is interconnected and carried out systematically to ensure that the data obtained during the benchmarking process can be used to fairly evaluate the capability of each web server and produce conclusions that are consistent with the objectives of the study.

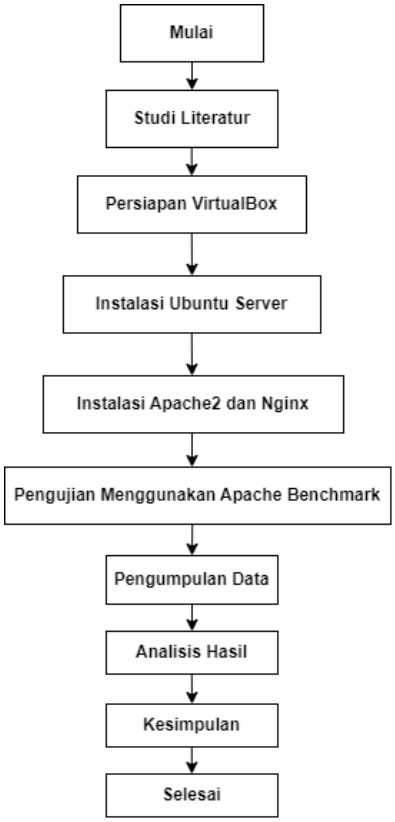

Figure 1: Research Stages

Based on Figure 1, the study begins by reviewing reference books to obtain information regarding Apache2, Nginx, Ubuntu Server, and Apache Benchmark. The next step is to prepare the virtualization environment using VirtualBox, followed by installing Ubuntu Server as the primary operating system. Subsequently, the Apache2 and Nginx web servers are installed and configured separately. After the configuration is completed, performance testing is conducted using Apache Benchmark with various numbers of requests and concurrent users. The test results are then collected and analyzed based on several parameters, including requests per second, time per request, failed requests, and transfer rate, so that conclusions can be drawn regarding the performance comparison between the two web servers.

3. Results and Discussion

3.1. Apache2 Results and Testing

Before the testing process was conducted, the Apache2 web server was first installed and configured on the Ubuntu Server operating system running in the VirtualBox virtualization environment. After the installation process was completed, a connectivity test was performed to ensure that the Apache2 service could be accessed through a web browser and was ready to receive client requests. Performance testing was then carried out using Apache Benchmark with several variations in the number of requests and concurrent users. This method was used to determine Apache2's capability in handling simultaneous client requests based on the parameters of requests per second, time per request, failed requests, and transfer rate [20].

The test was conducted by enabling the Apache2 service on Ubuntu Server and running Apache Benchmark through the Linux terminal. The benchmarking process was performed by specifying the same number of requests and concurrent users for each testing scenario. The test results were then recorded and analyzed based on the parameters of requests per second, time per request, failed requests, and transfer rate.Apache2 testing was performed using Apache Benchmark on an Ubuntu Server running in the VirtualBox virtualization environment. This test aimed to evaluate Apache2's capability in serving client requests based on several parameters, including requests per second, average time per request, failed requests, and data transfer rate [18].

The testing was conducted using various numbers of requests and concurrent users to determine the extent of Apache2's capability in handling client access loads gradually. The benchmarking process was carried out independently to maintain server stability and avoid interference with other web servers.Based on the testing results, Apache2 was able to handle all client requests successfully without any failed requests during every benchmarking process. This indicates that Apache2 can operate stably on Ubuntu Server throughout the testing process.

The requests per second value indicates the number of requests that the server is capable of processing in one second. Based on the obtained test results, Apache2 demonstrated good request processing capability in both low-request and high-request testing scenarios. However, an increase in the number of concurrent users affected the time per request value, causing the server response time to increase as the access load became higher. In addition, the transfer rate value indicates the amount of data successfully transmitted by the server to the client during the benchmarking process. Based on the test results, Apache2 was able to provide stable data transfer throughout the testing process [19].

The Apache2 testing results in this study indicate that Apache2 can still perform well on Ubuntu Server in handling client requests. However, an increase in the number of requests and simultaneously connected users may slow down the server response time because Apache2 uses a process-based and thread-based method to handle user connections.

The results of Apache2 testing using Apache Benchmark are shown in Figure 2.

```
gongmanis@ubuntu-server:~$ ab -n 100 -c 10 http://127.0.0.1/
his is ApacheBench, Version 2.3 <$Revision: 1923142 $>
opyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
icensed to The Apache Software Foundation, http://www.apache.org/

enchmarking 127.0.0.1 (be patient).....done


erver Software:        Apache/2.4.66
erver Hostname:        127.0.0.1
erver Port:            80

ocument Path:          /
ocument Length:        10672 bytes

oncurrency Level:      10
ime taken for tests:   0.129 seconds
omplete requests:      100
ailed requests:        0
otal transferred:      1094600 bytes
TML transferred:       1067200 bytes
equests per second:    773.26 [#/sec] (mean)
ime per request:       12.932 [ms] (mean)
ime per request:       1.293 [ms] (mean, across all concurrent requests)
ransfer rate:          8265.70 [Kbytes/sec] received

onnection Times (ms)
              min  mean[+/-sd] median   max
onnect:         0    0   0.6      0       6
rocessing:      0   11  13.1      6      69
aiting:         0    6   9.2      3      65
otal:           0   12  13.3      6      69

ercentage of the requests served within a certain time (ms)
 50%      6
 66%     13
 75%     18
 80%     20
 90%     33
 95%     43
 98%     46
 99%     69
100%     69 (longest request)
```

Figure 2: Apache2 test results using Apache Benchmark.

Based on Figure 2, Apache2 was able to process all submitted requests without experiencing any failed requests. An increase in the number of concurrent users caused the time per request value to increase gradually. This indicates that Apache2 requires greater resources when handling a large number of simultaneous connections.

3.2. Nginx Testing Results

Nginx testing was conducted using Apache Benchmark on Ubuntu Server running in the VirtualBox virtualization environment. This testing aimed to evaluate the performance of Nginx in handling client requests based on the parameters of requests per second, time per request, failed requests, and transfer rate.After the installation and configuration were completed, the Nginx web server was tested using Apache Benchmark with the same testing scenario as Apache2. The testing was conducted separately to avoid port usage conflicts and to

ensure consistent benchmarking results. The observed parameters included requests per second, time per request, failed requests, and transfer rate.

The testing process was carried out using various numbers of requests and concurrent users to determine the extent of Nginx's capability in handling client access loads gradually. The testing was conducted separately from Apache2 to maintain server stability and avoid port conflicts on Ubuntu Server.Based on the testing results presented in Chapter III above, Nginx was able to handle all client requests without experiencing any failed requests during every benchmarking process. This indicates that Nginx can operate stably throughout the testing process.

The requests per second value of Nginx was higher than that of Apache2. This indicates that Nginx is capable of processing a larger number of client requests within a shorter period of time. In addition, the time per request value of Nginx was also lower, resulting in faster server response times when receiving client requests [18], [20].The transfer rate value of Nginx indicates the server's capability to transfer data to clients during the benchmarking process. Based on the obtained test results, Nginx achieved a higher transfer rate than Apache2, resulting in more optimal server performance in handling client access.

After the installation and configuration process was completed, the Nginx web server was tested using Apache Benchmark with the same testing scenario as Apache2. The testing was conducted separately to avoid port usage conflicts and to maintain the consistency of the benchmarking results. The observed parameters included requests per second, time per request, failed requests, and transfer rate.

The testing results indicate that Nginx provides better performance in handling concurrent requests on Ubuntu Server [11]. This finding is consistent with the study in [12], which shows that Nginx can maintain stable server performance in a Linux environment due to its capability to manage connections efficiently. This is influenced by the event-driven architecture used by Nginx, which enables more efficient utilization of server resources when handling a large number of simultaneous client connections. This finding is also consistent with the study in [13], which demonstrates that Nginx is capable of distributing and handling client requests efficiently through an optimal connection management mechanism, thereby maintaining stable server performance when receiving increased access loads.

The performance testing results of Nginx using Apache Benchmark are shown in Figure 3.

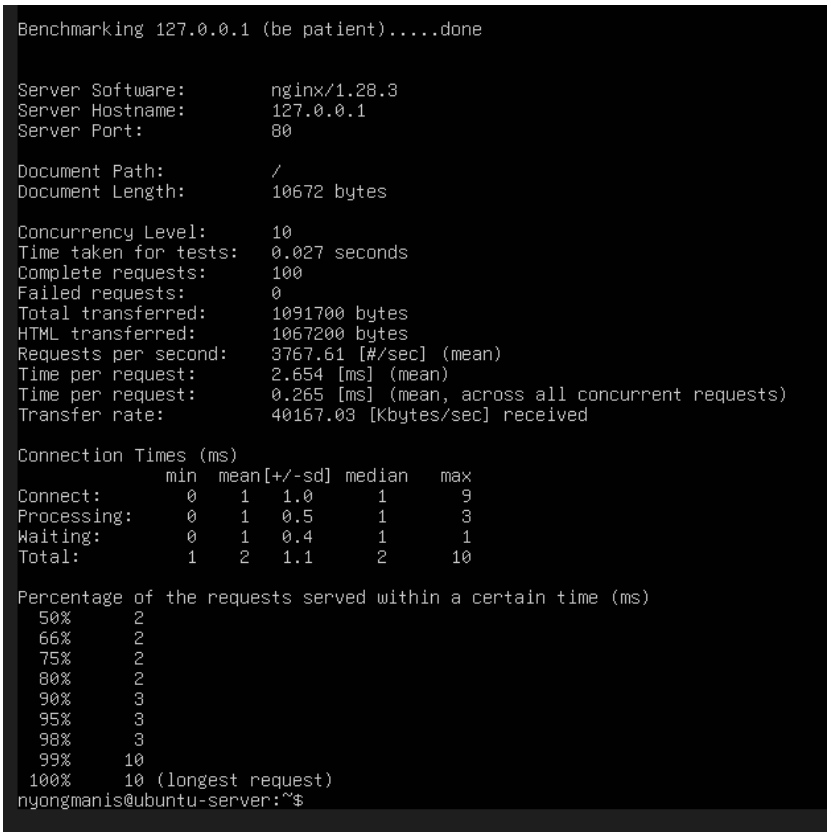

Figure 3: Nginx Testing Results Using Apache Benchmark

Based on Figure 3, Nginx was able to handle all submitted requests without experiencing any failed requests. In addition, the requests per second value achieved by Nginx was higher than that of Apache2, while the time per request value was lower, indicating that Nginx was able to provide faster response times.

3.3. Comparative Analysis of Apache2 and Nginx

Based on the testing results obtained using Apache Benchmark, performance differences were identified between the Apache2 and Nginx web servers on Ubuntu Server. The performance comparison can be observed through the parameters of requests per second, time per request, failed requests, and transfer rate obtained during the benchmarking process.The difference in performance between Apache2 and Nginx is determined by the operating mechanism and architecture used by each web server. Apache2 uses a process-based and thread-based processing model, in which each user request is handled by a separate process or thread. This mechanism provides considerable flexibility in meeting various web service requirements; however, system resource usage, such as memory and CPU, generally increases as the server receives more requests and connections.

The performance difference between Apache2 and Nginx is influenced by the architecture used by each web server. Apache2 adopts a process-based and thread-based approach, in which each client connection is handled by a specific process or thread. Meanwhile, Nginx uses an event-driven architecture that enables a single process to handle multiple connections simultaneously, making server resource utilization more efficient.The testing results show that Nginx achieved a higher requests per second value than Apache2. This indicates that Nginx is capable of processing a greater number of client requests per second. In addition, the time per request value of Nginx was lower than that of Apache2, resulting in faster server response times when handling client requests.

For the transfer rate parameter, Nginx achieved a higher data transfer rate than Apache2. This indicates that Nginx has a more optimal capability in transferring data to clients during the benchmarking process.Based on the testing results, both Apache2 and Nginx were able to operate stably without experiencing any failed requests throughout the benchmarking process. The performance of both web servers in maintaining service availability is also supported by the study in [15]. That study demonstrated that Apache2 and Nginx on the Ubuntu Linux operating system consistently provided stable service for valid requests, even when tested under various network traffic scenarios.

Apache2 uses a process-based and thread-based approach to handle client connections, resulting in greater server resource utilization when processing a large number of simultaneous requests. This condition causes the time per request value to increase as the number of concurrent users grows. According to the study, Apache's performance is affected by an increase in concurrent requests, which gradually increases server response time.

Meanwhile, Nginx implements an event-driven architecture that enables the server to manage a large number of client connections simultaneously with more efficient resource utilization. As a result, Nginx is able to deliver superior performance in concurrent request testing compared with Apache2. This finding is consistent with the study conducted by [17], which indicates that Nginx demonstrates superior performance compared to Apache in a Debian environment when analyzed using the WRK benchmarking method.

The results of this study are consistent with previous research stating that Nginx has better capability in handling concurrent connections and a large number of requests compared with Apache2 [9], [10].The findings of this study are also consistent with the previous investigation conducted by [16], which indicates that each web server has a unique performance profile. Nginx has proven to excel in handling large volumes of requests, while Apache continues to provide reliable service capabilities according to server configuration requirements.

To facilitate the comparative analysis of the performance of both web servers, the Apache2 and Nginx testing results based on the testing parameters are summarized in Table 1.

Table 1. Comparison

Parameter	Apache2	Nginx	Description
Requests per Second	Lower	Higher	Nginx performs better
Time per Request	Higher	Lower	Nginx is faster
Failed Requests	0	0	Same
Transfer Rate	Lower	Higher	Nginx performs better

Based on Table 1, Nginx demonstrated better performance than Apache2 across most of the testing parameters. Nginx achieved higher requests per second and transfer rate values, as well as a lower time per request value. Meanwhile, both web servers demonstrated good stability, as neither experienced failed requests during the benchmarking process.

3.4. Discussion and Comparison with Previous Studies

The results of this study indicate that Nginx provides better performance than Apache2 based on the parameters of requests per second, time per request, and transfer rate. These findings are consistent with the study conducted by [3], which states that Nginx has better efficiency in handling high access loads on the Linux operating system. The study conducted by [9] also showed that Nginx is capable of achieving higher requests per second values than Apache.

The study conducted by [4] states that the event-driven architecture of Nginx enables the server to handle a large number of concurrent connections with more efficient resource utilization. These findings are consistent with the results of this study, in which Nginx achieved a lower time per request value than Apache2.Nevertheless, Apache2 still demonstrated stable performance throughout the testing process and did not experience any failed requests. This indicates that Apache2 can still be used as a web server for website services with specific requirements, particularly in environments that require configuration flexibility and comprehensive module support [22].

Overall, the results of this study reinforce previous research indicating that Nginx is superior in handling high access loads, while Apache2 remains a suitable choice for website service implementation that requires compatibility and configuration flexibility.

4. Conclusion

Based on the series of experiments conducted in the Ubuntu Server environment using the Apache Benchmark method, the results indicate that both tested web servers, Apache2 and Nginx, were able to provide stable client request services without any failed requests across all testing scenarios. The testing, which involved variations in the number of requests and concurrent users, revealed significant performance differences in several key parameters, particularly requests per second, time per request, and transfer rate. Nginx demonstrated higher request processing capability and faster response times than Apache2, resulting in more efficient service under increasing access loads. This condition is related to the differences in the architectures used, where Nginx, with its event-driven approach, is able to manage a large number of simultaneous connections with more efficient resource utilization, whereas Apache2, which is based on a process-based and thread-based architecture, tends to require greater system resources when handling many concurrent connections. Nevertheless, Apache2 still demonstrated good stability in handling all requests without errors, indicating that it remains suitable for environments with specific requirements that emphasize configuration flexibility and comprehensive module support. Overall, the results of this study indicate that the selection of a web server should be adjusted according to workload characteristics and system requirements, where Nginx is more recommended for high-traffic services, while Apache2 is suitable for environments requiring greater flexibility and lower traffic demands. For future research, it is recommended to conduct testing using higher server specifications, larger workload variations, and additional testing methods such as stress testing and CPU and RAM usage monitoring to obtain a more comprehensive and in-depth performance analysis of each web server.

References

[1] K. M. D. Pertiwi, R. F. Maulana, and A. Y. Wicaksono, "Implementasi dan Analisa Performa Berbagai Platform Web Server pada Lingkungan Multicore," Jurnal Edukasi dan Penelitian Informatika, vol. 8, no. 2, pp. 275–283, 2022.

[2] Y. N. Khamidah, B. Arifwidodo, and M. A. Amanaf, "Pengaruh Client Request Pada Web Server Apache dan Nginx Dengan IPv6 Menggunakan Apache Benchmark," Techno (Jurnal Fakultas Teknik Universitas Muhammadiyah Purwokerto), vol. 25, no. 1, 2024, doi: 10.30595/techno.v25i1.14056.

[3] W. Wicoksono, H. A. Mustaqhim, P. P. Anwas, and L. N. L. Badratul, "Performance Comparison of NGINX, Apache, and Lighttpd Using WRK on a Debian," Bit-Tech, vol. 8, no. 1, pp. 670–680, 2025, doi: 10.32877/bt.v8i1.2661.

[4] H. A. Rafiu, O. S. Adesina, and K. S. Adekeye, "Modelling the Response Rate of Apache Web Server Using Extreme Value Theory," Scientific African, vol. 23, Art. no. e02086, 2024, doi: 10.1016/j.sciaf.2024.e02086.

[5] S. Akva, S. Akva, Y. Lorinez, and J. Tampubolon, "Analisis Penggunaan Load Balancing dengan HAProxy terhadap Kinerja Web Server pada Ubuntu," Jurnal Multimedia dan Teknologi Informasi, vol. 7, no. 1, pp. 165–171, 2025.

[6] R. J. L. Purba, Y. S. M. Gurning, and A. A. Silaen, "Analisis Performa Web Server Apache Menggunakan Sistem Operasi Arch Linux dan Archcraft Linux," Journal of Humanities Education Management Accounting and Transportation, vol. 2, no. 1, pp. 480–497, 2025, doi: 10.57235/hemat.v2i1.5110.

[7] T. Takagaki and K. Mizutani, "Deep Analysis of Containerized Web Server Performance Using eBPF-Captured Data," IEEJ Transactions on Electrical and Electronic Engineering, vol. 20, no. 11, pp. 1826–1828, 2025, doi: 10.1002/tee.70070.

[8] T. Prantl, L. Wolf, J. Schiller, M. Pruckner, S. Kounev, T. Dänzer, and J. von Kistowski, "Server-Side Performance and Environmental Impact Evaluation of a Hybrid E-Learning Website," Discover Applied Sciences, 2025, doi: 10.1007/s42452-025-08059-x.

[9] F. K. Ramadhan, G. Garno, and A. Solehudin, "Comparative Study of Web Server Performance Testing with and without Docker Based on Virtual Machines," Journal of Applied Informatics and Computing, vol. 8, no. 1, pp. 155–166, 2024, doi: 10.30871/jaic.v8i1.3884.

[10] A. P. G. Perdana, "Analisis Perbandingan Performa Web Server Apache Menggunakan Docker dan Podman," Jurnal Informatics and Computer Science (JINACS), vol. 6, no. 4, pp. 1145–1155, 2025.

[11] R. Januhandini and U. Y. Oktiawati, "Analisis Kinerja Web Server Apache, Nginx, dan Caddy dengan Metode Stress Testing Menggunakan Autocannon," Journal of Internet and Software Engineering (JISE), vol. 6, no. 2, pp. 96–100, 2024.

[12] M. Naufal, A. Azi, B. Arifwidodo, and E. Wahyudi, "Analisis Performansi Web Server Saat Menangani Permintaan Client Menggunakan Metode Reverse Proxy Caching Nginx dan Varnish," Open Access Journal of Telecommunication, Electronics, and Control Engineering (JTECE), vol. 5, no. 1, pp. 14–21, 2023.

[13] M. L. Cahyadi, H. Setiawan, and Z. R. Mair, "Analisis Perbandingan Kinerja Web Server Nginx dan LiteSpeed Menggunakan Httperf dengan Sistem Operasi Debian," Jurnal Pengembangan Sistem Informasi dan Informatika, vol. 4, no. 2, 2023.

[14] A. Algiffary and T. Sutabri, "Analisis Kinerja Algoritme WRR dengan WLC pada Load Balancing Nginx," Indonesian Journal of Computer Science, vol. 12, no. 2, pp. 284–301, 2023.

[15] Agustine, Lady, and S. Seimahuira, "Penerapan Metode SAW dalam Analisa Perbandingan Performa Web Server (Apache, Nginx, Lighttpd, IIS) pada Bahasa Pemrograman PHP," REMIK (Riset dan E-Jurnal Manajemen Informatika Komputer), vol. 7, no. 1, pp. 409–420, 2023.

[16] D. S. Rahmadani, T. Ariyadi, E. Novitasari, and D. Pertiwiningsih, "Analisis Kinerja Web Server Berbasis Linux Ubuntu 22.04," Jurnal Tera, vol. 5, no. 1, pp. 35–44, 2025, doi: 10.59832/jt.v5i1.348.

[17] S. D. Riskiono and D. Pasha, "Analisis Perbandingan Server Load Balancing dengan HAProxy & Nginx dalam Mendukung Kinerja Server E-Learning," Jurnal Telekomunikasi dan Komputer, vol. 10, no. 3, pp. 135–144, 2020, doi: 10.22441/incomtech.v10i3.8751.

[18] A. Zulfahrizan, M. R. A. Tanjung, M. Z. Al-Kautsar, and D. Kiswanto, "Analisis Perbandingan Performa MySQL pada Lingkungan Virtualisasi pada Linux Debian dan Ubuntu Berbasis Web Server Nginx," JATI (Jurnal Mahasiswa Teknik Informatika), vol. 9, no. 1, pp. 1107–1112, 2024, doi: 10.36040/jati.v9i1.12545.

[19] A. Syarif Aziz, "Analisis Kinerja Web Server Apache, Nginx, OpenLiteSpeed, dan OpenResty," Journal of Informatics and Computer Science, vol. 11, no. 1, pp. 1–7, 2025.

[20] H. Tang, S. S. Kolahi, and L. Thompson, "Analysis of HTTP DDoS Flood Attacks on Apache2 and Nginx Web Servers with Linux Ubuntu," in Proceedings of the International Seminar on Intelligent Technology and Its Applications (ISITIA), 2024, pp. 716–721, doi: 10.1109/ISITIA63062.2024.10668064.

[21] A. R. Busran, "Analisis Perbandingan Performa Apache Web Server dan Nginx Menggunakan Apache JMeter," Jurnal Teknoif, vol. 8, no. 2, pp. 87–92, 2020.