Performance Comparison of Web Server Application on Single Board Computer

1st Mega Pranata
S1 Teknik Informatika
Institut Teknologi Telkom Purwokerto
Purwokerto, Indonesia
mega@ittelkom-pwt.ac.id

2nd Aditya Wijayanto
S1 Rekayasa Perangkat Lunak
Institut Teknologi Telkom Purwokerto
Purwokerto, Indonesia
aditya.wijayanto@ittelkom-pwt.ac.id

3rd Muhammad Fajar Sidiq
S1 Teknik Informatika
Institut Teknologi Telkom Purwokerto
Purwokerto, Indonesia
fajar@ittelkom-pwt.ac.id

***Abstract*—This research aims to compare two Web Servers on Single board computer (SBC). Sbc is usually used for IoT applications because only needs a small amount of power and is compact in size. Raspberry Pi 4 Model B with 8GB of RAM and 128GB SD Card is used in this research. We compare Apache Web Server and Nginx as web servers. WordPress and Apache Benchmark are used to evaluate the performance of web servers. The servers were tested with 10, 25, 50, and 100 concurrency levels with 1000 requests every test.**

***Index Terms*—single board computer, apache, nginx, web server**

I. Introduction

Single board computer are becoming more affordable with the development of ARM Processor. Their productivity is increasing steadily. Most of them do not required forced cooling, which makes them quieter, more realible and energy-saving. Such a device consume about 5W on the average [1]. Raspberry Pi is small single board computer series.The Raspberry Pi 4 Model B is the first of a new generation of Raspberry Pi computers supporting more RAM and with siginficantly enhanced CPU, GPU and I/O performance. RPi4B is available with many size of RAM [2]. Raspberry pi is the low power consumtion, cheap price and it has been widely implemented in the development of IoT technology [3]. Raspberry Pi can run with many Linux OS, Raspberry Pi OS is one of them, that supported by Raspberry Pi.

Web servers development is always challenging. Servers need to perform efficiently for attending request from thousand of clients simultaneously. The performance of web servers plays a key role in a large and growing community of web users. Apache is open source web server that widely-used. Apache uses multi-proces architecture and multi-thread architecture. Nginx is second widely-used open source web server, uses asynchronous multi-process event driven architecture [4].

Thanks to Institut Teknologi Telkom Purwokerto for funding this research.

A content management system (CMS) is any system used to create and publicize digital content, including a platform for creating static sites, blogs, online stores, forums, and others in between. WordPress is one of the well-known CMSs including Joomla, Drupal, and all the others [5]. WordPress had 68.7% of the CMS technologies based on market share in 2022 [6]. WordPress recommends host supports PHP version 7.4 or greater, with Mysql 5.7 or greater or MariaDB version 10.3 or greater and HTTPS Support [7].

II. Experiment

This research used two Raspberry Pi 4 B with 8GB of RAM and 128GB of storage. Raspberry Pi OS is used for the operating system, one RPi4B using Apache and the other using Nginx. The server deploys a WordPress with PHP-FPM and MariaDB for the databases. The WordPress uses default theme and default test post without any changes. Apache benchmark used for generating the traffic and performing the test. Apache bencmark is a tool for benchmarking HTTP Server, this will show how many requests per second the server is capable to serve [8].

A raspberry pi OS was installed, new user account was added to the OS. The user is added to the sudoers list, so the user is capable to execute command that requires administrative privileges. Before starting the software installation, the first step was to update the software package. After the update is complete, on the first RPi4B install an apache HTTP server, and on the second RPi4B install the Nginx server. After installation is complete, test the web server by accessing the IP address of each server using a web browser.All web server comes with default configuration, no tweak was performed on this research.

The next step was to install the PHP and PHP-FPM to support the PHP programming language and related extensions. MariaDB was installed as a database service for the server. Specific user created for accessing the

database from WordPress. Create a database WordPress, and install WordPress on the server.

All WordPress file placed on /home/konten/web/wordpress directory. Every server has one VirtualHost with a custom domain, each wp1.nginx.ittp for the Nginx server and wp1.apache.ittp for the apache server.

Every server was tested with apache Benchmark. Test conducted with 10, 25, 50, and 100 concurrency levels with 1000 total complete requests. Each test scenario was repeated 10 times.

III. Result

A. 10 *Concurrency Levels*

The first test was conducted with 10 concurrency levels and repeated 10 times on each web server. Apache web server gets 14.23 requests per second for the lowest, 14.53 requests per second for the highest, and 14.45 requests on average requests per second. On the Time per request concurrent, apache gets $702.852ms$ for the highest result, $688.257ms$ for the lowest, and $692.189ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $70.285ms$ for the highest result, $68.826ms$ for the lowest, and $69.219ms$ for the average time per request. Table I shows the details result of the Apache web server with 10 concurrency levels.

TABLE I
APACHE WITH 10 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.23	702.852	70.285
2	14.53	688.257	68.826
3	14.42	693.49	69.349
4	14.47	690.969	69.097
5	14.46	691.434	69.143
6	14.47	691.227	69.123
7	14.46	691.426	69.143
8	14.49	690.218	69.022
9	14.49	690.225	69.022
10	14.46	691.796	69.18
Avg	14.45	692.189	69.219
Max	14.53	702.852	70.285
Min	14.23	688.257	68.826

The Nginx server gets a better value for each parameter. On the requests per second parameter, the Nginx server gets 14.56 requests per second for the lowest, 14.74 requests per second for the highest, and 14.62 requests on average requests per second. On the Time per request concurrent, Nginx gets $686.586ms$ for the highest result, $678.345ms$ for the lowest, and $683.845ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $68.659ms$ for the highest result, $67.835ms$ for the lowest, and $68.385ms$ for the average time per request. Table II shows the details result of the Nginx server with 10 concurrency levels.

TABLE II
NGINX WITH 10 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.69	680.651	68.065
2	14.74	678.345	67.835
3	14.59	685.207	68.521
4	14.63	683.594	68.359
5	14.64	682.955	68.295
6	14.6	684.913	68.491
7	14.59	685.281	68.528
8	14.58	686.08	68.608
9	14.6	684.842	68.484
10	14.56	686.586	68.659
Avg	14.62	683.846	68.385
Max	14.74	686.586	68.659
Min	14.56	678.345	67.835

B. 25 *Concurrency Levels*

The second test use 25 as the value concurrency levels and repeated 10 times on each web server. Apache web server gets 14.41 requests per second for the lowest, 14.56 requests per second for the highest, and 14.47 requests on average requests per second. On the Time per request concurrent, apache gets $1734.827ms$ for the highest result, $1716.579ms$ for the lowest, and $1727.5061ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $69.393ms$ for the highest result, $68.663ms$ for the lowest, and $69.1ms$ for the average time per request. Table III shows the details result of the Apache web server with 25 concurrency levels.

TABLE III
APACHE WITH 25 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.53	1720.681	68.827
2	14.56	1716.579	68.663
3	14.45	1730.585	69.223
4	14.45	1730.633	69.225
5	14.48	1726.182	69.047
6	14.48	1726.336	69.053
7	14.46	1729.302	69.172
8	14.44	1731.177	69.247
9	14.46	1728.759	69.15
10	14.41	1734.827	69.393
Avg	14.47	1727.5061	69.1
Max	14.56	1734.827	69.393
Min	14.41	1716.579	68.663

TABLE IV
NGINX WITH 25 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.7	1700.425	68.017
2	14.73	1697.539	67.902
3	14.6	1712.351	68.494
4	14.6	1712.201	68.488
5	14.6	1712.17	68.487
6	14.63	1709.241	68.37
7	14.62	1710.378	68.415
8	14.6	1712.714	68.509
9	14.57	1716.437	68.657
10	14.56	1716.685	68.667
Avg	14.621	1710.0141	68.4006
Max	14.73	1716.685	68.667
Min	14.56	1697.539	67.902

TABLE V
APACHE WITH 50 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.55	3437.403	68.748
2	14.54	3439.442	68.789
3	14.45	3460.255	69.205
4	14.44	3463.117	69.262
5	14.44	3461.719	69.234
6	14.47	3456.326	69.127
7	14.47	3455.318	69.106
8	14.45	3460.207	69.204
9	14.44	3463.031	69.261
10	14.44	3462.101	69.242
Avg	14.469	3455.8919	69.1178
Max	14.55	3463.117	69.262
Min	14.44	3437.403	68.748

The result of Nginx gets 14.56 requests per second for the lowest, 14.73 requests per second for the highest, and 14.62 requests on average requests per second. On the Time per request concurrent, Nginx gets $1716.685ms$ for the highest result, $1697.539ms$ for the lowest, and $1710.014ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $68.667ms$ for the highest result, $67.902ms$ for the lowest, and $68.401ms$ for the average time per request. Table IV shows the details result of the Nginx server with 25 concurrency levels.

C. 50 *Concurrency Levels*

The next test use 50 as the value concurrency levels and repeated 10 times on each web server. Apache web server gets 14.44 requests per second for the lowest, 14.55 requests per second for the highest, and 14.47 requests on average requests per second. On the Time per request concurrent, apache gets $3463.117ms$ for the highest result, $3437.403ms$ for the lowest, and $3455.8919ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $69.262ms$ for the highest result, $68.748ms$ for the lowest, and $69.1178ms$ for the average time per request. Table V shows the details result of the Apache web server with 50 concurrency levels.

The result of Nginx gets 14.53 requests per second for the lowest, 14.70 requests per second for the highest, and 14.61 requests on average requests per second. On the Time per request concurrent, Nginx gets $3441.672ms$ for the highest result, $3401.079ms$ for the lowest, and $13423.651ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $68.833ms$ for the highest result, $68.022ms$ for the lowest, and $68.473ms$ for the average time per request. Table 6 shows the details result of the Nginx server with 50 concurrency levels.

TABLE VI
NGINX WITH 50 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.7	3401.079	68.022
2	14.64	3415.013	68.3
3	14.6	3423.953	68.479
4	14.61	3422.832	68.457
5	14.53	3441.672	68.833
6	14.61	3423.396	68.468
7	14.61	3422.055	68.441
8	14.58	3428.608	68.572
9	14.58	3430.452	68.609
10	14.59	3427.449	68.549
Avg	14.605	3423.6509	68.473
Max	14.7	3441.672	68.833
Min	14.53	3401.079	68.022

D. 100 *Concurrency Levels*

The last test was conducted with 100 concurrency levels and repeated 10 times on each web server. Apache web server gets 14.41 requests per second for the lowest, 14.52 requests per second for the highest, and 14.45 requests on average requests per second. On the Time per request concurrent, apache gets $6937.797ms$ for the highest result, $6886.353ms$ for the lowest, and $6920.6371ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $69.378ms$ for the highest result, $68.864ms$ for the lowest, and $69.2063ms$ for the average time per request. Table VII shows the details result of the Apache web server with 100 concurrency levels.

The Nginx server gets 14.56 requests per second for the lowest, 14.73 requests per second for the highest, and 14.61 requests on average requests per second. On the

TABLE VII
APACHE WITH 100 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.45	6918.935	69.189
2	14.52	6886.353	68.864
3	14.47	6912.124	69.121
4	14.44	6925.281	69.253
5	14.42	6933.228	69.332
6	14.44	6925.133	69.251
7	14.42	6933.431	69.334
8	14.48	6907.416	69.074
9	14.41	6937.797	69.378
10	14.44	6926.673	69.267
Avg	14.449	6920.6371	69.2063
Max	14.52	6937.797	69.378
Min	14.41	6886.353	68.864

TABLE VIII
NGINX WITH 100 CONCURRENCY LEVELS

Test Number	Request per second	Time per request (Concurrent) (ms)	Time per request (ms)
1	14.73	6789.573	67.896
2	14.61	6846.044	68.46
3	14.62	6838.312	68.383
4	14.63	6836.08	68.361
5	14.57	6861.257	68.613
6	14.6	6849.807	68.498
7	14.6	6851.419	68.514
8	14.58	6857.163	68.572
9	14.58	6857.303	68.573
10	14.56	6868.05	68.681
Avg	14.608	6845.5008	68.4551
Max	14.73	6868.05	68.681
Min	14.56	6789.573	67.896

Time per request concurrent, Nginx gets $6868.050ms$ for the highest result, $6789.573ms$ for the lowest, and $6845.501ms$ on average time per request concurrent. The mean time per request across all concurrent requests results in $68.681ms$ for the highest result, $67.896ms$ for the lowest, and $68.455ms$ for the average time per request. Table VIII shows the details result of the Nginx server with 100 concurrency levels.

IV. CONCLUSION

The result shows the performance of Nginx is better than Apache web server. Figure 1 shows the average requests per second on each concurrency level, the result gives less than 0.1 difference on each level. The highest average requests per second were achieved by the Nginx server with 14.62, and the lowest average requests per second come from Apache with 14.45. Figure 2 shows the average time per request on every concurrency level, Nginx gives better performance with 68.38 ms average time per request than Apache with 69.22 ms average time per request. Figure 3 shows the average time per request (Concurrent), Nginx still overcomes apache on these parameters.

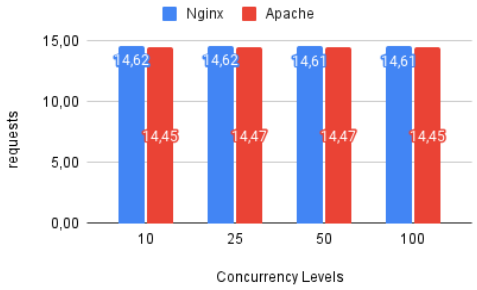

Fig. 1. Average Requests per Second.

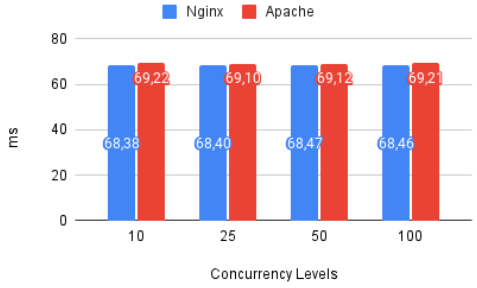

Fig. 2. Average Time per Request.

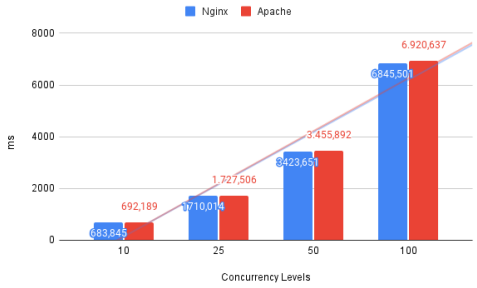

Fig. 3. Average Time per Request (Concurrent).

The results show the overall performance of Nginx comes better than Apache web server on RPi4B with 8GB of RAM.

ACKNOWLEDGMENT

This research is part of Electronic Election Using Single Board Computer Research, funded by Institut Teknologi Telkom Purwokerto.

REFERENCES

[1] . K. K., "SINGLE-BOARD SERVER AND SOFTWARE APPLICATION USED IN WEATHER STATION," Int. J. Res. Eng. Technol., vol. 03, no. 29, pp. 85–88, Dec. 2014, doi: 10.15623/ijret.2014.0329015.

[2] "Datasheet Raspberry Pi 4 Model B."

[3] Y.-C. Lee and C.-M. Lee, "Real-Time Smart Home Surveillance System of Based on Raspberry Pi," in 2020 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE), Yunlin, Taiwan, Oct. 2020, pp. 72–74. doi: 10.1109/ECICE50847.2020.9301929.

[4] Prakash P, Biju R, and M. Kamath, "Performance analysis of process driven and event driven web servers," in 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, India, Jan. 2015, pp. 1–7. doi: 10.1109/ISCO.2015.7282230.

[5] J. Cabot, "WordPress: A Content Management System to Democratize Publishing," IEEE Softw., vol. 35, no. 3, pp. 89–92, May 2018, doi: 10.1109/MS.2018.2141016.

[6] "CMS market share, websites and contacts - Wappalyzer." https://www.wappalyzer.com/technologies/cms (accessed Jul. 29, 2022).

[7] "Hosting Requirements for WordPress," WordPress.org. https://wordpress.org/about/requirements/ (accessed Jul. 29, 2022).

[8] "ab - Apache HTTP server benchmarking tool - Apache HTTP Server Version 2.4." https://httpd.apache.org/docs/2.4/programs/ab.html (accessed Jul. 30, 2022).