

Analisis Performansi Antara Apache & Nginx Web Server dalam Menangani *Client Request*

Albert Yakobus Chandra

Universitas Mercu Buana Yogyakarta

e-mail: albert.ch@mercubuana-yogya.ac.id

Diajukan: 16 November 2019; Direvisi: 27 November 2019; Diterima: 28 November 2019

Abstrak

Dalam keberadaan *World Wide Web* (WWW), *web server* menjadi salah satu faktor penting agar sebuah *website* dapat berjalan dengan baik dan melayani kebutuhan pengguna *web* tersebut. *Web server* yang tepat digunakan untuk sebuah sistem *website* maka dapat dipastikan *website* tersebut selalu dapat berjalan dengan baik. Saat ini banyak pilihan *web server* yang dapat digunakan untuk menjalankan sistem *website*, dua yang paling populer digunakan adalah Apache dan Nginx *web server*. Penelitian ini akan melakukan pengujian pada kedua *web server* tersebut untuk mengetahui salah satu *web server* yang terbaik dalam menyelesaikan *client request*. Pengujian yang dilakukan menggunakan tools Apache Bench untuk melakukan benchmarking dari sisi banyak nya *client request* yang bervariasi mulai dari 100 request sampai dengan 1000000 request dan waktu yang dibutuhkan untuk menyelesaikan request tersebut. Hasil uji pada penelitian ini memberikan hasil benchmarking yang menunjukkan bahwa dari sisi penggunaan waktu, nginx menggunakan waktu lebih sedikit daripada Apache dalam menyelesaikan *client request*.

Kata kunci: *Web server, Website, Benchmarking, Nginx, Apache.*

Abstract

In existence of the *World Wide Web* (WWW), *web server* is one of the required important factors for a *website* could run well and serve the user of the *website*. The right *web server* used for a *website* then the *website* could always run well without any problem. There are currently many *web servers* that could be used to run a *website* system, two of them that most popular is Apache and Nginx *web server*. This research will test the two *web servers* to find out which one *web server* is the best at response to the *client request*. The test will be using Apache Bench tool to carried out benchmark from the varied *client request* start from 100 request to 1000000 request and how many times that used to complete the request. The test result of this study provides the benchmark result which is show that Nginx using more less time in completing *client request* rather than apache.

Keywords: *Web server, Website, Benchmarking, Nginx, Apache.*

1. Pendahuluan

Keberadaan *World Wide Web* (WWW) sebagai media perantara untuk menyajikan *hypermedia* melalui jaringan internet telah menjadi sebuah kebutuhan. Di mana pertumbuhan pengguna telah menunjukkan peningkatan yang eksponensial dalam beberapa tahun terakhir. Beberapa penelitian menunjukkan bahwa lebih dari 75% lalu lintas data yang mengalir melalui jaringan internet didominasi oleh data *Hypertext Transfer Protocol* (HTTP) [1]. Oleh karena hal tersebut maka perencanaan dan penyediaan sumber daya *web server* untuk melayani permintaan berbasis HTTP tersebut menjadi hal yang sangat penting [2].

Dengan paradigma layanan WWW, maka layanan yang diberikan kepada pengguna (*end user*) akan berbasis pada paradigma *client-server*. *Client* adalah sisi aplikasi yang akan melakukan permintaan (*request*) ke server, permintaan dari *client* ini dapat berupa objek berbentuk teks, dokumen (*file*), video, audio, *database*, *query*, dan lain-lain. Sedangkan server adalah sisi aplikasi yang akan merespons permintaan dari *client* tersebut dan menyajikan data sesuai dengan *request* tersebut [2][3].

Server sendiri dalam tugasnya merespons permintaan dari *client* dibutuhkan *web server* untuk mengolah permintaan tersebut sebelum mengirimkan kembali jawaban yang disebut *response*. Apache, Nginx (Engine-x), LightHTTPD, Hiawata, Cherokee, Apache Tomcat merupakan contoh dari aplikasi-aplikasi *web server* yang digunakan di dunia [4]. Berdasarkan survei yang dilakukan oleh *Web Technology*

Surveys, Apache dan Nginx merupakan dua aplikasi web server yang paling banyak digunakan. Di mana Apache memiliki persentase kegunaan sebanyak 44,2% dibandingkan dengan Nginx yang memiliki persentase sebanyak 41,1% [5].

Beberapa penelitian sebelumnya dalam hal menguji *web server* seperti penelitian yang berjudul “Analisis *Web Server* untuk pengembangan *Hosting Server* Institusi: Perbandingan Kinerja *Web Server* Apache dan Nginx” hasil dari penelitian ini menyatakan bahwa Apache lebih unggul dibandingkan Nginx dari sisi *transfer rate* dan *connection time* [6]. selanjutnya penelitian berjudul “Analisis Perbandingan Performa *Web Server* Apache dan Nginx Menggunakan HTTPERF Pada VPS Dengan Sistem Operasi CentOS” hasil dari penelitian ini menyatakan bahwa Nginx lebih unggul dalam merespons pada subjek web statis sedangkan Apache lebih unggul dalam segi *bandwidth* [7].

Berbeda dari penelitian yang sudah pernah dilakukan sebelumnya penelitian ini akan melakukan pengujian antara dua *web server* yang paling banyak digunakan yaitu Apache dan Nginx [5] untuk mengetahui *web server* mana yang terbaik di antara kedua *web server* tersebut dalam memberikan *response* terhadap *request* dari *client*. Metode komparasi yang digunakan pada penelitian ini adalah *benchmark test* menggunakan *tool* dari Apache Bench di mana *tool* ini dapat mengukur bagaimana kinerja dari *web server* yang digunakan dan berapa banyak waktu yang digunakan dari *request per second* yang mampu di *servicing* oleh *web server* yang digunakan.

Pengujian kedua *web server* ini dilakukan dengan melakukan *setup* dan *installation* Apache dan Nginx pada *Hardware Environment* yang sama yaitu Dell PowerEdge R630, Xeon E5-2620, dengan menggunakan Proxmox VE sebagai *Virtual Environment* dan sistem operasi CentOS 7. Selanjutnya Apache Bench dijalankan untuk melakukan *benchmarking* dengan jumlah *request* dan beban *request* yang bervariasi mulai 100 *request* hingga 1000000 *request*.

2. Web Server dan *Bechmarking Tools*

2.1. Apache *Web Server*

Apache *Web Server* merupakan *unix-based web server*, Apache awalnya dikembangkan berbasis kode pada NCSA HTTPD 1.3 yang kemudian diprogram ulang menjadi sebuah *web server* yang paling banyak digunakan saat ini. Apache kini menjadi *web server* yang paling populer dan banyak digunakan lebih dari 42% dari berbagai domain *website* yang ada di internet [5]. Apache memiliki fitur yang sangat lengkap mulai dari performa yang tinggi, fungsionalitas, efisiensi, serta kecepatan. Apache juga merupakan *web server* berbasis open source [8].

2.2. Nginx *Web Server*

Nginx atau biasa disebut “Engine-x”, adalah *open source web server*[9]. Nginx selain digunakan sebagai *web server* juga memiliki fitur untuk digunakan sebagai *reverse proxy*, *HTTP cache*, dan *load balancer*[10]. Nginx dibuat oleh Igor Sysoev dan dirilis ke publik pada bulan Oktober 2004. Saat awal dirilis Igor meyakinkan publik bahwa Nginx dapat menjadi jawaban untuk mengatasi permasalahan yang ada pada saat itu yaitu permasalahan performa *web server* jika memiliki koneksi aktif lebih dari 10.000 koneksi secara bersamaan. Nginx menawarkan penggunaan memori yang lebih rendah dibandingkan web server lainnya dan juga beberapa fitur seperti: *reverse proxy*, IPv6, *load balancing*, *FastCGI support*, *web sockets*, *handling static files*, *TLS/SSL*.

2.3. Apache Bench

Apache Bench adalah sebuah *tool* dari Apache *organization* yang digunakan untuk mengukur performansi pada *Hypertext Transfer Protocol* (HTTP) *web server*. *Tool* ini digunakan untuk menghitung berapa banyak *request per second* yang dapat di layani oleh web server yang digunakan. Beberapa fitur dari Apache Bench seperti: *open source*, *simple command line*, *platform independent*, *load and performance test*, *not extensible*. Apache Bench dapat digunakan untuk menguji performa dari *web server* dengan berbagai batasan pengujian seperti *transfer rate* dan *request per second*.

2.4. CentOS

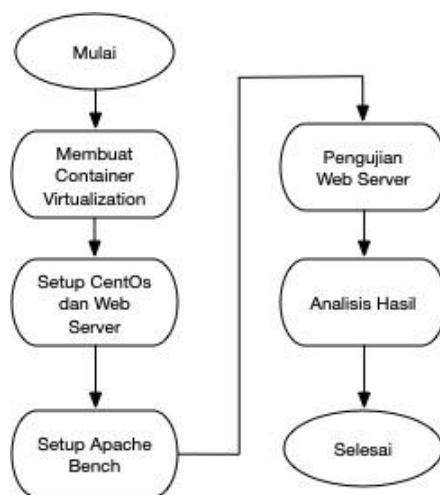
CentOS (*Community Enterprise Operating System*) merupakan sebuah sistem operasi berbasis Linux yang saat ini paling populer digunakan sebagai sistem operasi dari sebuah server. Centos menyediakan *platform* pada skala *enterprise* secara *free* (tidak berbayar) [11]. CentOS dasarnya dibuat dari *source code* Red Hat Enterprise, *software* ini dirilis di bawah *General Public License* (GPL) yang selanjutnya dikembangkan oleh komunitas yang disebut *CentOS Project*.

2.5. Container-Based Virtualization

Container-based virtualization adalah langkah terbaik dan mudah untuk membuat *virtual user space* [12], ini dilakukan dengan membagi *kernel* dari sistem operasi utama (*host*). *User space* dari sistem operasi utama atau *host* dibagi menjadi *isolated user space* yang disebut *container*. Model *container* ini tidak perlu menjalankan sistem operasi tamu (*guest*) dan *virtual machine* untuk setiap *virtualized environment*, sehingga model *container* ini memiliki kelebihan yaitu *overhead* yang lebih kecil dibandingkan dengan metode *hypervisor-based*. Selain itu *container-based virtualization* dapat memberikan *user space* yang lebih besar [12]. *Container-based virtualization* ini digunakan sebagai *container* untuk menjadi tempat yang akan menjalankan sistem operasi, *web server*, dan *benchmarking tools*.

3. Metode Penelitian

Metode penelitian yang digunakan pada penelitian ini adalah metode eksperimen, yaitu metode yang bertujuan untuk menguji pengaruh suatu variabel terhadap variabel lain, atau menguji hubungan sebab akibat antara variabel yang satu dengan lainnya. Metode penelitian eksperimen memiliki perbedaan yang jelas dibanding dengan metode penelitian lainnya, yaitu adanya pengontrolan terhadap variabel penelitian dan adanya pemberian perlakuan terhadap kelompok eksperimen [13][14]. Jalannya penelitian dapat dilihat pada gambar diagram jalan penelitian berikut ini:



Gambar 1. Diagram jalannya penelitian.

4. Hasil dan Pembahasan

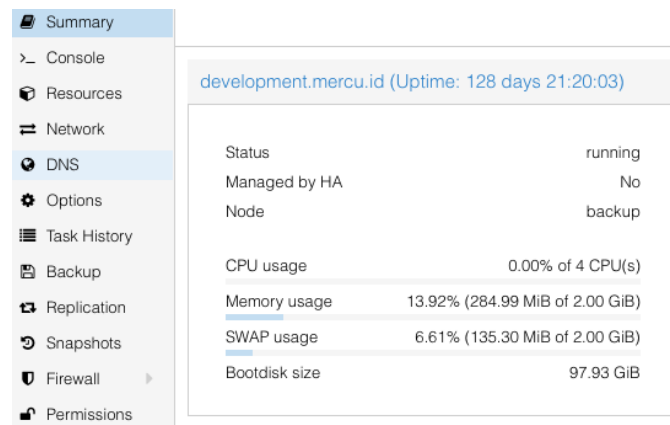
Berdasarkan diagram jalannya penelitian maka untuk melakukan analisis performansi pada *web server* dilakukan dengan langkah-langkah berikut:

4.1. Membuat Container Virtualization

Container virtualization digunakan sebagai *virtual space* dari sebuah server, *virtual space* ini digunakan untuk menjalankan sistem operasi, *web server*, dan juga Apache Bench yang merupakan *tool* untuk pengujian *web server*. Sistem aplikasi untuk membuat *Container* ini akan menggunakan Proxmox VE. *Virtual space container* pada penelitian ini akan memiliki spesifikasi sebagai berikut:

Tabel 1. Spesifikasi *virtual space container*.

No	Nama Hardware	Spesifikasi
1	Intel Xeon E5 Processor	4 Core
2	Memori	2 GB
3	Storage	100 GB
4	Network	Ethernet Gbits



Gambar 2. Spesifikasi *Virtual Space Container* pada ProxmoxVE.

4.2. Pengujian *Web Server*

Pengujian *web server* menggunakan Apache Bench dilakukan melalui *bash* dengan memberikan beban *request* bertahap mulai dari *range* 100, 1000, 10000, sampai 100000 *request*. Dari beban *request* yang diberikan kepada *web server* tersebut kemudian kita dapat melihat berapa lama waktu yang dihabiskan *web server* untuk menyelesaikan *request* tersebut.

4.2.1. Pengujian Apache *Web Server*

Pengujian pertama dilakukan pada Apache *web server*, dengan beban *request* 100 sampai 100000 yang diberikan secara bertahap.

a. 100 *request*

Perintah *bash* Apache Bench yang dieksekusi:

```
“ab -n 100 -c 100 http://103.58.111.134/”
```

```
[root@development ~]# ab -n 100 -c 100 http://103.58.111.134/
This is ApacheBench, Version 2.3 <$Revision: 1430300 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 103.58.111.134 (be patient).....done

Server Software:      Apache/2.4.6
Server Hostname:     103.58.111.134
Server Port:        80

Document Path:      /
Document Length:    4897 bytes

Concurrency Level:   100
Time taken for tests: 0.007 seconds
Complete requests:  100
Failed requests:     0
Write errors:        0
Non-2xx responses:  100
Total transferred:  516800 bytes
HTML transferred:   489700 bytes
Requests per second: 13726.84 [#/sec] (mean)
Time per request:   7.285 [ms] (mean)
Time per request:   0.073 [ms] (mean, across all concurrent requests)
Transfer rate:      69277.63 [Kbytes/sec] received
```

Gambar 3. Hasil pengujian Apache 100 *request*.

Dari hasil pengujian pada Gambar 3 dapat kita lihat bahwa Apache *web server* dalam mengeksekusi 100 *request* membutuhkan waktu 0.007 *seconds*.

b. 1000 *request*

Perintah *bash* Apache Bench yang dieksekusi:

```
“ab -n 1000 -c 100 http://103.58.111.134/”
```

```

Benchmarking 103.58.111.134 (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests

Server Software:      Apache/2.4.6
Server Hostname:      103.58.111.134
Server Port:          80

Document Path:        /
Document Length:      4897 bytes

Concurrency Level:    100
Time taken for tests:  0.059 seconds
Complete requests:    1000
Failed requests:       0
Write errors:         0
Non-2xx responses:    1000
Total transferred:    5168000 bytes
HTML transferred:     4897000 bytes
Requests per second:  16964.39 [#/sec] (mean)
Time per request:     5.895 [ms] (mean)
Time per request:     0.059 [ms] (mean, across all concurrent requests)
Transfer rate:        85617.16 [Kbytes/sec] received

```

Gambar 4. Hasil pengujian Apache 1000 *request*.

Dari hasil pengujian pada Gambar 4 dapat kita lihat bahwa Apache *web server* dalam mengeksekusi 1000 *request* membutuhkan waktu 0.059 *seconds*.

c. 10000 *request*

Perintah *bash* Apache Bench yang dieksekusi:

“ab -n 10000 -c 100 http://103.58.111.134/”

```

Benchmarking 103.58.111.134 (be patient)
Completed 1000 requests
Completed 2000 requests
Completed 3000 requests
Completed 4000 requests
Completed 5000 requests
Completed 6000 requests
Completed 7000 requests
Completed 8000 requests
Completed 9000 requests
Completed 10000 requests
Finished 10000 requests

Server Software:      Apache/2.4.6
Server Hostname:      103.58.111.134
Server Port:          80

Document Path:        /
Document Length:      4897 bytes

Concurrency Level:    100
Time taken for tests:  0.434 seconds
Complete requests:    10000
Failed requests:       0
Write errors:         0
Non-2xx responses:    10000
Total transferred:    51680000 bytes
HTML transferred:     48970000 bytes
Requests per second:  23028.79 [#/sec] (mean)
Time per request:     4.342 [ms] (mean)
Time per request:     0.043 [ms] (mean, across all concurrent requests)
Transfer rate:        116223.44 [Kbytes/sec] received

```

Gambar 5. Hasil pengujian Apache 10000 *request*.

Dari hasil pengujian pada Gambar 5 dapat kita lihat bahwa Apache *web server* dalam mengeksekusi 10000 *request* membutuhkan waktu 0.434 *seconds*.

- d. 100000 *request*
Perintah *bash* Apache Bench yang dieksekusi:
“ab -n 100000 -c 100 http://103.58.111.134/”

```
Benchmarking 103.58.111.134 (be patient)
Completed 10000 requests
Completed 20000 requests
Completed 30000 requests
Completed 40000 requests
Completed 50000 requests
Completed 60000 requests
Completed 70000 requests
Completed 80000 requests
Completed 90000 requests
Completed 100000 requests
Finished 100000 requests

Server Software:      Apache/2.4.6
Server Hostname:     103.58.111.134
Server Port:         80

Document Path:       /
Document Length:     4897 bytes

Concurrency Level:   100
Time taken for tests: 4.095 seconds
Complete requests:   100000
Failed requests:     0
Write errors:        0
Non-2xx responses:   100000
Total transferred:   516800000 bytes
HTML transferred:    489700000 bytes
Requests per second: 24422.61 [#/sec] (mean)
Time per request:    4.095 [ms] (mean)
Time per request:    0.041 [ms] (mean, across all concurrent requests)
Transfer rate:       123257.87 [Kbytes/sec] received
```

Gambar 6. Hasil pengujian Apache 100000 *request*.

Dari hasil pengujian pada Gambar 6 dapat kita lihat bahwa Apache *web server* dalam mengeksekusi 100000 *request* membutuhkan waktu 4.095 *seconds*.

4.2.2. Pengujian Nginx Web Server

Pengujian pertama dilakukan pada Nginx *web server*, dengan beban *request* 100 sampai 100000 yang diberikan secara bertahap.

- a. 100 *request*
Perintah *bash* Apache Bench yang dieksekusi:
“ab -n 100 -c 100 http://103.58.111.134/”

```
Benchmarking 103.58.111.134 (be patient).....done

Server Software:      nginx/1.12.2
Server Hostname:     103.58.111.134
Server Port:         80

Document Path:       /
Document Length:     3700 bytes

Concurrency Level:   100
Time taken for tests: 0.005 seconds
Complete requests:   100
Failed requests:     0
Write errors:        0
Total transferred:   393400 bytes
HTML transferred:    370000 bytes
Requests per second: 21598.27 [#/sec] (mean)
Time per request:    4.630 [ms] (mean)
Time per request:    0.046 [ms] (mean, across all concurrent requests)
Transfer rate:       82976.17 [Kbytes/sec] received
```

Gambar 7. Hasil pengujian Nginx 100 *request*.

Dari hasil pengujian pada Gambar 7 dapat kita lihat bahwa Nginx *web server* dalam mengeksekusi 100 *request* membutuhkan waktu 0.005 *seconds*.

- b. 1000 request
Perintah *bash* Apache Bench yang dieksekusi:
“ab -n 1000 -c 100 http://103.58.111.134/”

```
Benchmarking 103.58.111.134 (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests

Server Software:      nginx/1.12.2
Server Hostname:     103.58.111.134
Server Port:         80

Document Path:       /
Document Length:     3700 bytes

Concurrency Level:   100
Time taken for tests: 0.035 seconds
Complete requests:   1000
Failed requests:     0
Write errors:        0
Total transferred:   3934000 bytes
HTML transferred:    3700000 bytes
Requests per second: 28568.98 [#/sec] (mean)
Time per request:    3.500 [ms] (mean)
Time per request:    0.035 [ms] (mean, across all concurrent requests)
Transfer rate:       109756.22 [Kbytes/sec] received
```

Gambar 8. Hasil pengujian Nginx 100 request.

Dari hasil pengujian pada Gambar 8 dapat kita lihat bahwa Nginx *web server* dalam mengeksekusi 1000 request membutuhkan waktu 0.035 seconds.

- c. 10000 request
Perintah *bash* Apache Bench yang dieksekusi:
“ab -n 10000 -c 100 http://103.58.111.134/”

```
Benchmarking 103.58.111.134 (be patient)
Completed 1000 requests
Completed 2000 requests
Completed 3000 requests
Completed 4000 requests
Completed 5000 requests
Completed 6000 requests
Completed 7000 requests
Completed 8000 requests
Completed 9000 requests
Completed 10000 requests
Finished 10000 requests

Server Software:      nginx/1.12.2
Server Hostname:     103.58.111.134
Server Port:         80

Document Path:       /
Document Length:     3700 bytes

Concurrency Level:   100
Time taken for tests: 0.351 seconds
Complete requests:   10000
Failed requests:     0
Write errors:        0
Total transferred:   39340000 bytes
HTML transferred:    37000000 bytes
Requests per second: 28449.50 [#/sec] (mean)
Time per request:    3.515 [ms] (mean)
Time per request:    0.035 [ms] (mean, across all concurrent requests)
Transfer rate:       109297.21 [Kbytes/sec] received
```

Gambar 9. Hasil pengujian Nginx 100 request.

Dari hasil pengujian pada Gambar 9 dapat kita lihat bahwa Nginx *web server* dalam mengeksekusi 10000 *request* membutuhkan waktu 0.351*seconds*.

- d. 100000 *request*
Perintah bash apache bench yang dieksekusi:
“ab -n 100000 -c 100 http://103.58.111.134/”

```

Benchmarking 103.58.111.134 (be patient)
Completed 10000 requests
Completed 20000 requests
Completed 30000 requests
Completed 40000 requests
Completed 50000 requests
Completed 60000 requests
Completed 70000 requests
Completed 80000 requests
Completed 90000 requests
Completed 100000 requests
Finished 100000 requests

Server Software:      nginx/1.12.2
Server Hostname:      103.58.111.134
Server Port:          80

Document Path:        /
Document Length:      3700 bytes

Concurrency Level:    100
Time taken for tests:  3.420 seconds
Complete requests:    100000
Failed requests:       0
Write errors:          0
Total transferred:    393400000 bytes
HTML transferred:     370000000 bytes
Requests per second:  29237.86 [#/sec] (mean)
Time per request:     3.420 [ms] (mean)
Time per request:     0.034 [ms] (mean, across all concurrent requests)
Transfer rate:         112325.92 [Kbytes/sec] received
    
```

Gambar 10. Hasil pengujian Nginx 100 *request*.

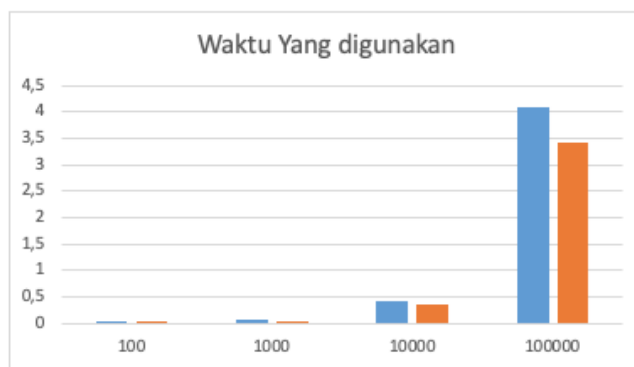
Dari hasil pengujian pada gambar 10 dapat kita lihat bahwa nginx *web server* dalam mengeksekusi 100000 *request* membutuhkan waktu 3.420 *seconds*.

4.3. Analisis Hasil

Hasil pengujian yang dilakukan terhadap kedua *web server* yaitu Apache dan Nginx dapat dilihat pada tabel hasil pengujian berikut:

Tabel 2. Waktu yang digunakan antara Apache dan Nginx.

Web Server	Jumlah Request	Waktu yang digunakan (Seconds)
Apache	100	0,007
	1000	0,059
	10000	0,434
	100000	4,095
Nginx	100	0,005
	1000	0,035
	100000	3,420



Gambar 11. Chart Hasil Pengujian. Biru:Apache, Orange: Nginx.

Dari Tabel 2 hasil pengujian dan *chart* pada Gambar 11 dapat kita lihat bahwa Nginx memiliki rata-rata waktu penyelesaian *request* yang lebih cepat dibandingkan dengan Apache sehingga dapat diambil kesimpulan bahwa performansi Nginx lebih baik dibandingkan dengan Apache dalam hal menangani *client request*.

5. Kesimpulan

Penelitian ini dilakukan dengan harapan dapat memberikan hasil uji performansi pada kedua *web server* yang populer digunakan saat ini yaitu Apache dan Nginx dalam menangani *client request*. Dari hasil uji yang dilakukan pada kedua *web server* tersebut menunjukkan bahwa Nginx memiliki rata-rata waktu penyelesaian *request* yang lebih cepat dibandingkan dengan Apache. Hasil ini didapatkan setelah proses pengujian dengan jumlah *request* mulai dari 100 sampai 1000000 dengan menggunakan *tool* Apache Bench.

Berdasarkan atas hasil uji yang didapatkan pada penelitian ini, maka diharapkan dapat memberikan masukan sebagai pilihan *web server* mana yang akan digunakan nantinya dalam membangun sebuah sistem *website* yang akan digunakan jika *website* tersebut akan menangani banyaknya *client request*. Kemudian berdasarkan hasil yang diperoleh maka penelitian ini dapat dilanjutkan dengan menguji performansi kedua *web server* tersebut dari aspek lain seperti *concurrency level* dan jumlah *url* diakses pada waktu yang bersamaan.

Daftar Pustaka

- [1] K. Claffy, G. Miller, and K. Thompson, "The Nature of the Beast: Recent Traffic Measurements from an Internet Backbone," *Int. Netw. Conf. '98, Geneva, Switzerland, Jul 1998, Internet Soc.*, vol. et'98, no. February 2016, pp. 21–24, 1998.
- [2] K. Kant and Y. Won, "Server capacity planning for Web traffic workload," *IEEE Trans. Knowl. Data Eng.*, vol. 11, no. 5, pp. 731–747, 1999.
- [3] G. Li, H. Zheng, and G. Li, "Building a secure web server based on OpenSSL and apache," *Proc. Int. Conf. E-bus. E-Government, ICEE 2010*, pp. 1307–1310, 2010.
- [4] "Seven of the best open source web servers for your organisation | Gallery | Computerworld UK." [Online]. Available: <https://www.computerworlduk.com/galleries/open-source/best-open-source-web-servers-for-your-organisation-3655327/>. [Accessed: 12-Feb-2019].
- [5] "Apache vs. Nginx vs. Microsoft-IIS usage statistics, February 2019." [Online]. Available: <https://w3techs.com/technologies/comparison/ws-apache,ws-microsoftiis,ws-nginx>. [Accessed: 12-Feb-2019].
- [6] A. Aziz and T. Tampati, "Analisis Web Server untuk Pengembangan Hosting Server Institusi: Perbandingan Kinerja Web Server Apache dengan Nginx," *Multinetics*, vol. 1, no. 2, p. 12, 2015.
- [7] F. Adnan, "Analisis Perbandingan Performa Web Server Apache dan Nginx menggunakan Httpperf pada VPS dengan Sistem Operasi CentOS," *Stmik Amikom Yogyakarta*, p. 6, 2016.
- [8] Y. Hu, A. Nanda, and Q. Yang, "Measurement, analysis and performance improvement of the Apache Web server," *1999 IEEE Int. Performance, Comput. Commun. Conf. IPCCC 1999*, pp. 261–267, 1999.
- [9] X. Chi, B. Liu, Q. Niu, and Q. Wu, "Web load balance and cache optimization design based nginx under high-concurrency environment," *Proc. - 2012 3rd Int. Conf. Digit. Manuf. Autom. ICDMA 2012*, pp. 1029–1032, 2012.
- [10] M. Data, M. Luthfi, and W. Yahya, "Optimizing single low-end LAMP server using NGINX reverse proxy caching," *Proc. - 2017 Int. Conf. Sustain. Inf. Eng. Technol. SIET 2017*, vol. 2018-Janua, no. February, pp. 21–23, 2018.
- [11] The CentOS Project, "CentOS," 2019. [Online]. Available: <https://www.centos.org/about/>.
- [12] Y. Tachibana, J. Kon, and S. Yamaguchi, "A study on the performance of web applications based on RoR in a highly consolidated server with container-based virtualization," *Proc. - 2017 5th Int. Symp. Comput. Networking, CANDAR 2017*, vol. 2018-Janua, pp. 580–583, 2018.
- [13] N. S. Sukmadinata, *Metode Penelitian Pendidikan*. Bandung: Remaja Rosdakarya, 2009.
- [14] M. Sholikhin *et al.*, "Analisis Performansi Server Cluster Pada Load Balancing," pp. 1–8, 2016.