

PERBANDINGAN PERFORMANSI WEB SERVER APACHE DAN NGINX DENGAN MENGGUNAKAN IPV6

I Kadek Susila Satwika, Ketut Ngurah Semadi

Program Studi Teknik Informatika, STIMIK STIKOM Indonesia Denpasar, Indonesia

Email: susila.satwika@stiki-indonesia.ac.id

Abstrak. Seiring dengan meningkatnya pertumbuhan jumlah website dan aplikasi online, berdampak pada kesiapan resource pendukungnya. Seperti aplikasi web server yang digunakan serta ketersediaan IP. Web server harus memiliki performansi yang handal guna mendukung availability dari website dan aplikasi. 2 aplikasi Web server yang terkenal dan banyak digunakan adalah NGINX dan APACHE. Selain performansi server dibutuhkan juga ketersediaan IP yang berfungsi sebagai alamat identik suatu website atau aplikasi. Saat ini masih banyak digunakan IP version 4 (IPv4) sebagai alamat identik website dan aplikasi. Namun pesatnya pertumbuhan teknologi berbasis IP, menyebabkan ketersediaan IPv4 semakin berkurang dan terancam habis. Akibat permasalahan itu maka dikembangkan teknologi IP version 6 (IPv6) dengan jumlah alamat yang jauh lebih banyak dan memiliki keamanan yang lebih baik dari IPv4. Dalam penelitian ini menguji web server dengan aplikasi NGINX dan APACHE, dimana menggunakan IPv6 sebagai alamatnya. Penelitian ini mengukur performansi server dengan parameter Time Taken for Tests, Request Per Second, Transfer Rate (Kb/s), Time per Request (ms), Memory Usage, dan Load Maximum menggunakan aplikasi Apache Benchmarking. Dari hasil pengujian didapatkan hasil bahwa kinerja web server menggunakan NGINX memiliki performansi lebih baik dibandingkan dengan APACHE. Kemudian dari sisi penggunaan memory dan load pada server NGINX lebih efisien dibandingkan APACHE.

Kata kunci : web server, IPv6, NGINX, APACHE

Tidak dapat dipungkiri lagi bahwa peran Website saat ini sangat penting sekali. Tidak heran mengapa pertumbuhan jumlah Website sangat meningkat, khususnya website e-commerce [1]. Dengan memiliki kelebihan yang dapat diakses dari kapanpun dan dimanapun, Website sudah menjadi bagian khususnya bagi dunia usaha. Website memiliki kegunaan yaitu untuk mencari informasi mengenai perdagangan, berita, Pendidikan, informasi usaha (profile company), dan lain-lain.

Dengan meningkatnya jumlah Website tiap tahunnya maka resource untuk Website itu sendiri juga meningkat. Server merupakan tempat untuk menyimpan konten website yang sering disebut dengan hosting. Tanpa adanya server ini maka website tidak akan dapat diakses. Resource yang dimaksud dalam hal ini adalah Server serta alamat IP. Banyak aplikasi web server yang ada saat ini seperti Apache, Nginx (Engine-x), LightHTTPD, Hiawata, Cherokee, Apache Tomcat [2]. Namun berdasarkan survey yang telah dilakukan Web Technology Surveys, aplikasi NGINX dan APACHE merupakan aplikasi yang paling banyak

diguakan [3]. Sehingga dalam pengujian ini Server menggunakan aplikasi web Server APACHE atau NGINX.

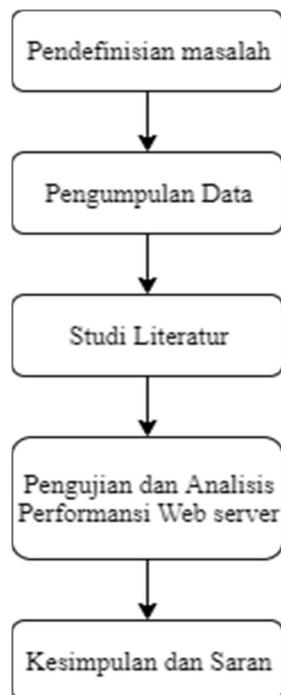
Kemudian Internet Protocol (IP) merupakan sebuah nomer unik yang nantinya sebagai alamat identik dari sebuah Website itu sendiri. IP akan memastikan sebuah domain yang diakses akan mengarah ke Server yang benar. Kendala yang terjadi adalah jumlah IPv4 saat ini sudah semakin berkurang. Dengan meningkatnya pertumbuhan teknologi berbasis IP, menyebabkan berkurangnya alokasi IP. Karena keterbatasan jumlah pengalamatan menggunakan IPv4, maka dikembangkan suatu standar baru yang mampu mengakomodasi jumlah pengalamatan yang lebih banyak. IP version 6 (IPV6) kemudian dijadikan standar baru yang kelak akan mengakomodasi pengalamatan IP yang lebih banyak. Untuk saat ini penerapan IPV6 dapat diintegrasika dengan IPV4 sehingga tidak perlu mengubah banyak konfigurasi infrastruktur eksisting [4] [5].

Terdapat beberapa penelitian yang telah melakukan pengujian mengenai web server APACHE dan NGINX, namun masih menggunakan IPV4 [6] [7]. Sehingga dalam penelitian ini melakukan sebuah analisis

performansi web *Server* NGINX dan APACHE dengan menggunakan IPv6. Web *Server* akan dikonfigurasi pada *virtual mechine* (Virtual Box) menggunakan OS Ubuntu 18.04. Kemudian akan dilakukan pengujian *Time Taken for Tests*, *Request Per Second*, *Transfer Rate* (Kb/s), *Time per Request* (ms), *Memory Usage*, dan *Load Maximum*. Diharapkan nantinya penelitian ini akan menjadi referensi dalam mengembangkan web *Server* dengan menggunakan IPv6.

I. Metodologi

Penelitian dilakukan dengan beberapa tahapan dimulai dari definisi masalah hingga tahapan pengujian dan pengambilan keputusan. Penelitian terbagi ke dalam beberapa langkah yang ditunjukkan seperti pada gambar 1



Gambar 1 Alur Penelitian

Pada penelitian ini membangun 2 buah *web Server* yang akan diinstall dengan aplikasi NGINX dan APACHE. Adapun *Server* yang digunakan adalah *virtual Server* dengan menggunakan aplikasi *virtual box*. Kemudian untuk masing-masing *Server* akan dikonfigurasi dengan menggunakan IPv6 sebagai alamat identik dari *Server* virtual. Pada penelitian ini juga melakukan pengukuran performansi *Server* dengan parameter *load*

average, *response time*, dan *request loss*. Nantinya akan dilakukan perbandingan performansi web *Server* antara web *Server* dengan aplikasi NGINX dengan APACHE. Adapun spesifikasi yang akan digunakan dari masing *Server* dapat dilihat pada Tabel 1.

Tabel 1. Spesifikasi *Server*

No	Parameter	Spesifikasi
1	<i>OS</i>	Ubuntu 18.04
2	<i>Memory</i>	2 GB
3	<i>Harddisk</i>	40 GB

Pengujian dilakukan dengan menggunakan aplikasi *apache benchmarking*. Aplikasi tersebut memberikan koneksi sesuai dengan yang kita inginkan ke *web Server*. Koneksi ini dapat dianalogikan jumlah pengunjung yang mencoba untuk mengakses *web Server*. Pada penelitian ini diberikan koneksi sebanyak 1.000, 10.000, dan 50.000, dengan jumlah *concurrency* (jumlah permintaan yang dilakukan sekaligus) sebanyak 1000 permintaan untuk masing-masing *web Server*. Setelah itu akan dipantau dan diukur performansi *Server* dengan parameter :

- *Time Taken for Tests*
- *Request Per Second*
- *Transfer Rate* (Kb/s)
- *Time per Request* (ms)
- *Memory Usage*
- *Load Maximum*

Apache Benchmarking Tool

Apache Benchmarking Tool adalah sebuah alat yang dibuat oleh Apache Organization yang dapat bekerja untuk mengukur performansi pada Hypertext Transfer Protocol (HTTP) *web Server*. *Tool* ini dapat menghitung banyak *request per second* yang dapat dilayani 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 Benchmarking Tool* dapat digunakan untuk menguji performa dari *web Server* dengan berbagai batasan pengujian seperti *transfer rate* dan *request per second* [8].

Untuk mengukur performansi web *Server* menggunakan Apache Benchmarking Tool dapat menggunakan perintah berikut :

```
ab -n 100 -c 10 http://[Address]/
```

- Parameter *n* adalah jumlah koneksi yang dibuat ke *Server* tujuan, dengan contoh diatas berarti koneksi yang dibuat adalah 100 koneksi.
- Parameter *c* adalah jumlah request *concurrent* (berbarengan) yang dibuat, dengan contoh diatas berarti jumlah request yang dibuat adalah 10 *request* dalam satu waktu.
- Parameter terakhir adalah *Address*. *Address* dapat berupa alamat IP atau halaman yang akan diproses oleh *web Server* di-*benchmark*.

Gambar 2 menunjukkan contoh pengujian menggunakan Apache Benchmarking Tool.

```
Benchmarking 216.244.67.176 (be patient)...
Server Software:      Apache/2.4.18
Server Hostname:     216.244.67.176
Server Port:         80

Document Path:       /
Document Length:     11321 bytes

Concurrency Level:   10
Time taken for tests: 31.061 seconds
Complete requests:   100
Failed requests:     0
Total transferred:   1159500 bytes
HTML transferred:   1132100 bytes
Requests per second: 3.22 [#/sec] (mean)
Time per request:    3106.148 [ms] (mean)
Time per request:    310.615 [ms] (mean,
Transfer rate:       36.45 [Kbytes/sec] r

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:  250  305 174.1   266  1312
Processing: 266 2631 628.1  2469 4594
Waiting:   266 1532 793.7  1547 3453
Total:     547 2936 621.6  2750 4859

Percentage of the requests served within a c
50%  2750
66%  2781
75%  3484
80%  3687
90%  3781
95%  3828
98%  4703
99%  4859
100% 4859 (longest request)
```

Gambar 2. Contoh hasil pengujian Apache Benchmarking Tool.

Apache Web Server

Apache Web Server merupakan *unix-based web server*. Apache merupakan *web server* yang paling populer dan banyak digunakan lebih dari 42% dari berbagai domain *website* yang ada di internet [3]. Apache pertama kali dikembangkan berbasiskode pada NCSA HTTPD 1.3 yang diprogram ulang menjadi sebuah *web server*. Apache memiliki fitur yang sangat lengkap mulai dari performa yang tinggi, fungsionalitas, efisiensi, serta kecepatan. Apache juga merupakan web server berbasis *open source* [9].

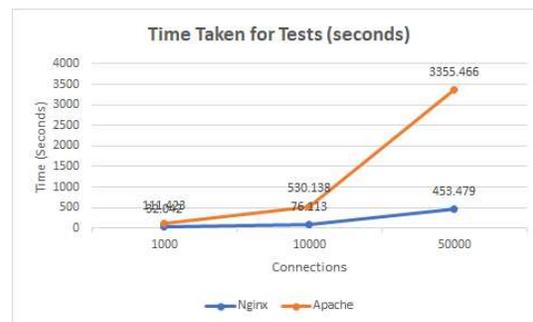
Nginx Web Server

Nginx atau biasa disebut “Engine-x”, adalah *open source web server* [10]. Nginx dibuat oleh Igor Sysoev dan dirilis ke publik pada bulan Oktober 2004. 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. Nginx selain digunakan sebagai web server juga memiliki fitur untuk digunakan sebagai *reverse proxy*, HTTP *cache*, dan *load balancer* [11].

II. Hasil dan Pembahasan

Pengukuran kinerja *Server* dilakukan dengan melakukan penguran kinerja *server* dengan parameter

- *Time Tken for Tests*
- *Request Per Second*
- *Transfer Rate (Kb/s)*
- *Time per Request (ms)*
- *Memory Usage*
- *Load Maximum*

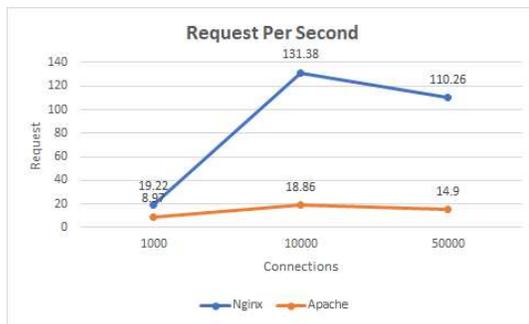


Gambar 3. Grafik Hasil *Time taken for Tests*

Gambar 3 menunjukkan hasil pengujian *Time taken for Tests*. Pengukuran ini bertujuan

untuk mengukur seberapa lama waktu yang dibutuhkan untuk melakukan satu kali pengujian. Pada gambar 3 terlihat semakin banyak jumlah koneksi menuju ke server maka waktu yang dibutuhkan menyelesaikan semua request juga semakin meningkat. Pada gambar 3 juga menunjukkan waktu yang dibutuhkan oleh server dengan menggunakan NGINX lebih cepat dibandingkan APACHE.

Gambar 4 menunjukkan hasil pengujian jumlah request per detik. Berdasarkan hasil pengujian terlihat semakin banyak jumlah koneksi tidak mengakibatkan peningkatan pada jumlah requestnya. Terlihat pada hasil pengujian pada webserver NGINX pada pengujian 10.000 koneksi lebih besar dibandingkan dengan 50.000 koneksi. Hal ini bergantung pada kemampuan atau beban kerja server. Jika pada suatu waktu beban kerja server meningkat, maka jumlah request yang mampu dilayani dapat berkurang. Selanjutnya pada gambar 4 juga terlihat web server dengan menggunakan NGINX mampu menangani jauh lebih banyak request dibandingkan dengan APACHE.



Gambar 4. Grafik Hasil Request Per Second

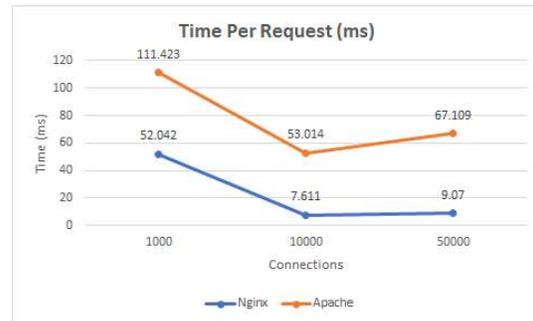


Gambar 5. Grafik Hasil Transfer Rate

Gambar 5 menunjukkan hasil pengujian Transfer Rate. Berdasarkan hasil pengujian terlihat bahwa kedua pengujian menghasilkan transfer rate tertinggi pada pada pengujian

10000 koneksi. Kemudian jika dibandingkan antara kedua server, server dengan menggunakan NGINX nilai transfer rate nya lebih tinggi dibandingkan dengan APACHE. Ini artinya NGINX memiliki kecepatan yang lebih tinggi dibandingkan APACHE. Hal ini dikarenakan NGINX mampu menangani jumlah request yang lebih banyak dibandingkan APACHE.

Gambar 6 menunjukkan hasil pengujian time per request. Pengujian ini mengukur rata-rata waktu yang dibutuhkan untuk server dalam menangani 1 buah request. Berdasarkan hasil pengujian terlihat semakin banyak jumlah request pada server NGINX menghasilkan waktu per request semakin kecil. Namun pada server APACHE waktu per request pada 10000 koneksi lebih kecil dibandingkan 50000 koneksi. Selain itu server NGINX memiliki waktu rata-rata yang jauh lebih kecil dibandingkan server APACHE. Hal ini menunjukkan server NGINX mampu menangani request lebih cepat dibandingkan dengan APACHE

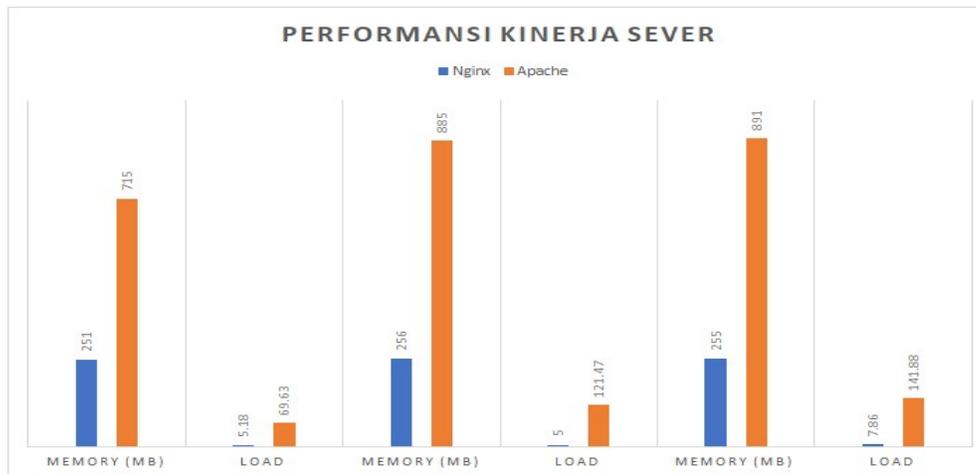


Gambar 6. Grafik Hasil Time per Request

Tabel 2. Hasil Pengukuran Memory dan Load Server

Server		Nginx	Apache
1000	Memory (MB)	251	715
	Load	5.18	69.63
10000	Memory (MB)	256	885
	Load	5	121.47
50000	Memory (MB)	255	891
	Load	7.86	141.88

Dari pengukuran pada sisi server dilakukan pengukuran memory dan load pada

Gambar 7. Grafik Hasil Pengukuran *Memory* dan *Load* Server

server. Berdasarkan tabel 6 dapat dilihat penggunaan *memory* tertinggi pada *server* NGINX terjadi pada kondisi 10000 koneksi dengan jumlah penggunaan *memory* maksimal 256 MB sedangkan penggunaan *memory* tertinggi pada *server* APACHE terjadi pada kondisi 50000 koneksi dengan jumlah penggunaan *memory* maksimal 891 MB.

Dari sisi penggunaan penggunaan *load server*, *server* NGINX memiliki *load* maksimal 7.86 request tiap menitnya pada kondisi 50000 koneksi. Sedangkan pada *server* APACHE *load* maksimal 141.88 request tiap menitnya pada kondisi 50000 request.

Berdasarkan Gambar 7, dapat dilihat perbandingan kinerja *server* NGINX dan APACHE. Gambar tersebut menunjukkan *server* NGINX memiliki efisiensi penggunaan *memory* dan kemampuan menangani request lebih baik dibandingkan dengan APACHE. Terlihat dari rata-rata penggunaan *memory* dan *load* pada *server* NGINX lebih kecil dibandingkan dengan *server* APACHE.

III. Kesimpulan

Penelitian ini telah melakukan pengukuran kinerja web *server* NGINX dan APACHE dengan menggunakan IPv6 dengan parameter *Time Taken for Tests*, *Request Per Second*, *Transfer Rate* (Kb/s), *Time per Request* (ms), *Memory Usage*, dan *Load Maximum*.

Dari hasil pengujian didapatkan hasil bahwa kinerja web *server* dengan menggunakan NGINX memiliki performansi lebih baik dibandingkan dengan APACHE.

Hal ini didapatkan berdasarkan parameter *Time Taken for Tests*, *Request Per Second*, *Transfer Rate* pada *server* NGINX menghasilkan hasil yang lebih baik dibandingkan APACHE. Kemudian dari sisi penggunaan *memory* dan *load* pada *server* NGINX lebih efisien dibandingkan APACHE.

IV. Daftar Pustaka

- [1] Warta Ekonomi, "Pertumbuhan E-Commerce Pesat di Indonesia," 19 Februari 2019. [Online]. Available: <https://www.wartaekonomi.co.id/read216302/pertumbuhan-e-commerce-pesat-di-indonesia.html>. [Accessed 16 November 2019].
- [2] Seven of the best open source web servers for your organisation | Gallery | Computerworld UK," computerworld, 30 April 2018. [Online]. Available: <https://www.computerworld.com/article/3412266/seven-of-the-best-open-source-web-servers-for-your-organisation.html>. [Accessed 10 Desember 2019].
- [3] Web Technology Survey. [Online]. Available: <https://w3techs.com/technologies/comparison/ws-apache>. [Accessed 10 Desember 2019].
- [4] I. Marzuki. (2018). Mekanisme Transisi IPv4 dan IPv6 Menggunakan Metode Automatic Tunneling Pada Jaringan Client Server Berbasis Linux. Jurnal Teknologi Informasi Indonesia (JTII), vol. Vol. 3 No. 2, no. Vol 3 No 2 (2018):

- Jurnal Teknologi Informasi Indonesia (November), pp. 68-73.
- [5] I. Warman and M. Y. S. N. Nugraha. (2017). Analisa Implementasi Interkoneksi Antara Ipv4 Dengan Ipv6 Menggunakan Metode Dual Stack Pada Mikrotik Routeros (Studi Kasus : Pt. Linggo Daya Energi). Jurnal Teknoif (Teknik Informatika), Vols. Vol 5, No 2, pp. 63-72.
- [6] I. F. Irza, Zuhendra and Efrizon. (2017). Analisis Perbandingan Kinerja Web Server Apache dan Nginx Menggunakan Httpperf Pada Portal Berita (Studi Kasus beritalinux.com). Jurnal Vocational Teknik Elektronika dan Informatika, Vols. Vol 5, No 2, pp. 75-82.
- [7] A. Y. Chandra. (2019). Analisis Performansi Antara Apache & Nginx Web Server dalam Menangani Client Request. Jurnal SIstem dan Informatika (JSI), vol. Vol 14 No 1 <https://doi.org/10.30864/jsi.v14i1.248>, pp. 48-56.
- [8] F. Adnan and Kusnawi. (2016). Analisis Perbandingan Performa Web Server Apache Dan Nginx Menggunakan Httpperf Pada Vps Dengan Sistem Operasi Centos. repository.amikom.ac.id, p. http://repository.amikom.ac.id/files/Publikasi_12.11.6594.pdf.
- [9] Y. Hu, A. Nanda and Q. Yang. (1999). Measurement, analysis and performance improvement of the Apache Web server. 1999 IEEE Int. Performance, Comput. Commun. Conf. IPCCC 1999, pp. 261–267.
- [10] X. Chi, B. Liu, Q. Niu and Q. Wu. (2012). Web load balance and cache optimization design based nginx under high-concurrency environment. in 3rd Int. Conf. Digit. Manuf. Autom. ICDMA 2012, pp.1029–1032,
- [11] M. Data, M. Luthfi and W. Yahya. (2017). Optimizing single low-end LAMP server using NGINX reverse proxy caching. Proc. - Int. Conf. Sustain. Inf. Eng. Technol. SIET 2017.