

Perbandingan Kinerja Web Server Terhadap Performa Integrasi Multi Basis Data

Dimas Pradana Putra¹, Triawan Adi Cahyanto², Reni Umilasari³,
Hardian Oktavianto⁴, Amalina Maryam Zakiyyah⁵

*dimaspradanap@gmail.com*¹, *triawan@unmuhjember.ac.id*², *reni.umilasari@unmuhjember.ac.id*³,
*hardian@unmuhjember.ac.id*⁴, *amalinamaryam@unmuhjember.ac.id*⁵

^{1,2,3,4,5} Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember

Keywords:

Web Service,
Nginx,
Apache,
Latency.

ABSTRACT

The research is focused for the purpose of analyzing the performance of which web server is able to receive more requests with a short connection time to support the performance of the integration application and can simulate the user with the jmeter application to test the web server. The performance test process is carried out using the benchmark test method using the Jmeter application by adjusting the input scenario and then analyzing the results based on the TIPHON standard. The results obtained from several variations of test parameters are known that the server's ability to send responses with the largest bytes is the apache web server, while the lowest error rate is nginx, while in terms of server network infrastructure, the apache web server is better because it can send a faster response. good and stable compared to nginx. Based on the test results above, it can be concluded that the apache web server has more effective results in web service performance against multi-database integration.

Kata Kunci

Web Service,
Nginx,
Apache,
Latency.

ABSTRAK

Penelitian difokuskan untuk tujuan menganalisa kinerja web *server* mana yang mampu menerima *request* lebih banyak dengan waktu koneksi yang singkat untuk menunjang kinerja aplikasi integrasi dan dapat mensimulasikan user dengan aplikasi jmeter untuk menguji web *server*. Proses uji kinerja dilakukan dengan metode *benchmark test* menggunakan aplikasi Jmeter dengan menyesuaikan skenario inputan kemudian menganalisa hasil berdasarkan standarisasi *TIPHON*. Hasil yang didapat dari beberapa variasi parameter pengujian diketahui kemampuan server untuk mengirim *response* dengan byte paling besar adalah *web server apache*, sedangkan *error rate* paling rendah dimiliki oleh *nginx*, sementara dari sisi infrastruktur jaringan server, *web server apache* lebih baik karena dapat mengirimkan *response* yang lebih baik dan stabil dibandingkan *nginx*. Berdasarkan hasil pengujian diatas, maka dapat disimpulkan bahwa *web server apache* memiliki hasil lebih efektif dalam kinerja web service terhadap integrasi multi basis data.

Korespondensi Penulis:

Dimas Pradana Putra
Universitas Muhammadiyah Jember
Jalan Karimatan No. 49, Jember, East Java 68121
Telepon: +62852-1237-0501
Email: *dimaspradanap@gmail.com*

1. PENDAHULUAN

Komponen penting dalam aplikasi berbasis web sebuah fasilitas jaringan yaitu *web server*. Tujuan *web server* diharapkan dapat memberikan kinerja yang maksimal dalam segala pengguna. Setiap *web server* mempunyai ciri khas dan karakteristik dalam mengatasi jumlah *request* dan *response*. Besarnya data yang diolah dapat membuat kinerja sistem semakin terbebani. Besarnya data yang diolah dapat membuat kinerja sistem semakin terbebani. Seperti integrasi data bidang tanah antara basis data Badan Pendapatan Pengelolaan Keuangan dan Aset Daerah (BPPKAD)

dan Badan Pertanahan Negara (BPN) yang per tahunnya menerima rata - rata 6.000 (enam ribu) transaksi proses jual beli atau peralihan hak atas tanah. Maka dari itu dukungan sistem yang baik dengan kinerja yang efektif sangat dibutuhkan mengingat banyaknya data yang diolah.

Berdasarkan latar belakang tersebut diatas, server BPPKAD yang menggunakan *APACHE* sebagai web servernya, memerlukan evaluasi terhadap kinerja penggunaan *web service* dalam mengintegrasikan basis data Bea Perolehan Hak atas Tanah dan Bangunan (BPHTB) dan basis data bidang tanah untuk mengukur tingkat efektifitas kinerja *web service* yang dipakai dengan melakukan *benchmarking* terhadap beberapa *web server* dengan menggunakan *metode benchmarking test*. Hasil yang didapat akan dianalisis untuk menentukan *web server* mana yang paling baik digunakan untuk menyokong proses integrasi data.

Beberapa penelitian sebelumnya dengan judul “*Kinerja Web Service pada Proses Integrasi Data*” membandingkan kinerja tiga pemodelan *web service* yang berjalan diatas tiga jenis *web server* menggunakan satu parameter transaksi, dari penelitian tersebut ditemukan bahwa konfigurasi server “*nginx*” dan klien “*apache*” dengan model *web service* “*REST*” merupakan konfigurasi yang paling baik untuk diimplementasikan dalam proses integrasi data [6]. Selanjutnya penelitian berjudul “*Analisis Performansi Antara Apache & Nginx Web Server dalam Menangani Client Request*” menyatakan bahwa *web server 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 [7]. Penelitian selanjutnya dengan judul “*Analisis Perbandingan Web Server Apache dan Nginx menggunakan Apache Jmeter pada Server Aplikasi AIS yang Berbasis Clustering Server*” mendapatkan hasil dari segi *throughput* dan *connect time* *web server nginx* lebih unggul dibanding *web server apache* [3]. Berikutnya penelitian dengan judul “*Analisa Perbandingan Kinerja Web Server Apache, Nginx, dan Litespeed Dengan Menggunakan Metode Stress Test*” dengan mengukur *throughput*, *error*, *sent*, *receive*, log dan *data encryption* *web server apache* dan *litespeed* menjadi *web server* yang paling baik saat penelitian dilakukan [8].

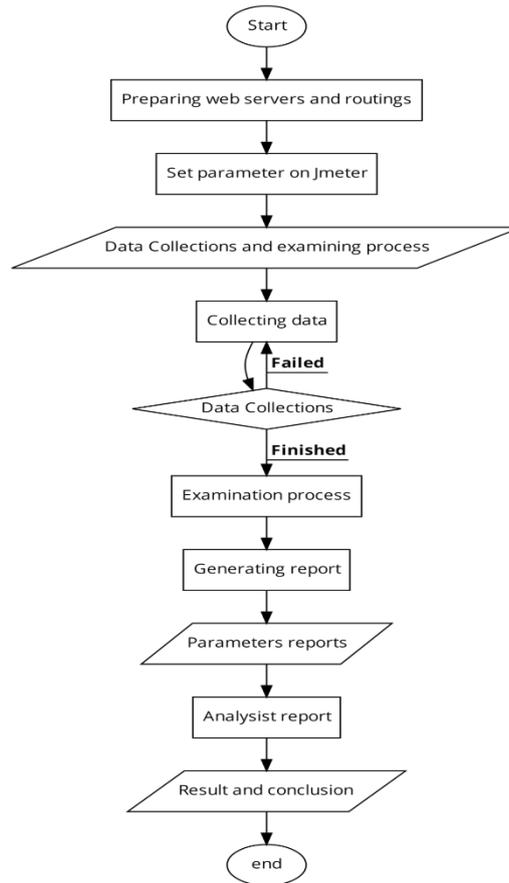
Berbeda dari beberapa penelitian sebelumnya, penelitian ini akan melakukan pengujian menggunakan 2 *web server* yang paling banyak digunakan yaitu *Apache* dan *Nginx* untuk mengetahui *web server* mana yang paling baik digunakan dalam memberikan *respons* terhadap *request* dari *client*. Metode komparasi yang digunakan pada penelitian ini adalah *benchmark test* menggunakan tool dari *Jmeter* 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 dilayani oleh *web server* yang digunakan.

Adapun perumusan masalah dapat diidentifikasi sebagai berikut: 1) Berapa hasil analisa dari 5 (lima) parameter *bytes throughput over time*, *connect time over time*, *response code per-second*, *response time distribution*, dan *latency* pada *web server nginx* dan *apache*?, dan 2) *Web server* mana yang lebih efisien untuk menyokong pengintegrasian data antara basis data BPPKAD dan BPN?

2. METODE PENELITIAN

Penelitian ini dilakukan di kantor Badan Pendapatan, Pengelolaan Keuangan dan Aset Daerah Kabupaten Situbondo. Perangkat keras dan perangkat lunak sangat dibutuhkan sebagai media penelitian untuk mendapatkan hasil dari 50, 100, 500, 1000, dan 5000 *Threads*. Penelitian ini menggunakan *Docker Container* yang berjalan diatas perangkat *server* dengan spesifikasi 1) *Intel(R) Xeon(R) CPU E3-1220 v6 @ 3.00GHz*, 2) *8 GB Memory*, 3) *1TB HDD*, sementara pengujiannya menggunakan aplikasi *JMeter* yang berjalan diatas perangkat *client* dengan spesifikasi 1) *Intel(R) Core(TM) i7-9750H*, 2) *16 GB Memory*, 3) *1 TB HDD*. Selanjutnya *Jmeter* dijalankan untuk melakukan *benchmarking* dengan jumlah request dan beban request yang bervariasi mulai 50 request hingga 5000 request

Penelitian ini membandingkan dua metode yang bisa digunakan untuk membuat sistem *web service* yaitu *Nginx* dan *Apache* dengan pemodelan *REST*. Sehingga dengan perbandingan ini bisa dilihat mana yang lebih efektif. Berdasarkan penelitian diharapkan mendapatkan parameter atau nilai yang biasanya digunakan sebagai parameter uji kinerja *web service* yaitu *bytes throughput over time*, *connect time over time*, *response code per-second*, *response time distribution*, dan *latency*. Perubahan kelima nilai dari parameter tersebut sangat tergantung dari *request* atau pemanggilan layanan *web service*. Berikut alur penelitian yang digambarkan dalam flowchart:



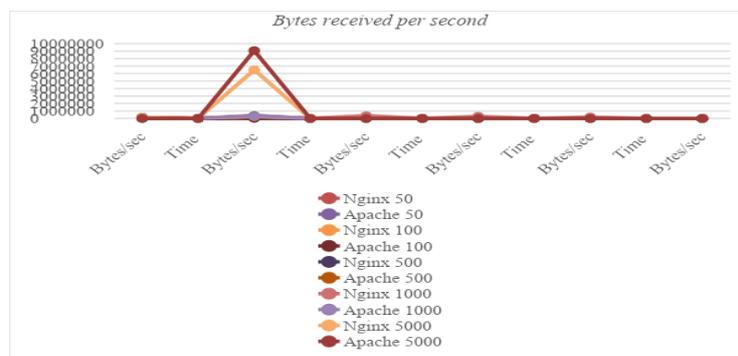
Gambar 1. Blok Diagram Pengukuran Efektivitas

3. HASIL DAN ANALISIS

Berdasarkan penelitian melakukan uji kinerja (performance) sistem dari dua metode yang digunakan untuk membuat sistem *web service* yaitu *Nginx* dan *Apache* dengan menggunakan lima parameter yaitu *bytes throughput over time*, *connect time over time*, *response code per-second*, *response time distribution*, dan *latency*. Berikut penjelasan mengenai hasil dan pembahasan:

3.1 Bytes Throughput Over Time

Throughput jaringan merupakan jumlah data yang dipindahkan dari suatu tempat ke tempat lain dalam periode waktu tertentu. *Bytes received per second* adalah penghitung *byte* yang diterima per detik melaporkan total *byte* yang diterima dari host per detik [3]. Berikut akan dijelaskan hasil pengujian *bytes received per second* dapat dilihat pada grafik dan tabel dibawah ini:



Gambar 2. Bytes received per second

Pada grafik ini menggambarkan *bytes received per second* secara garis besar hasil pengujian *web server nginx* respon tertinggi yang diterima pada *threads* 50 sebesar 41.548 *bytes/sec*, *threads* 100 sebesar 86.762

bytes/secthreads 500 sebesar 356.824 *bytes/sec*, *threads 1000* sebesar 355.602 *bytes/sec* dan *threads 5000* sebesar 6.490.022 *bytes/sec*. Sedangkan *web server apache* respon tertinggi yang diterima pada *threads 50* sebesar 44.840 *bytes/sec*, *threads 100* sebesar 84.660 *bytes/sec*, *threads 500* sebesar 206.670 *bytes/sec*, *threads 1000* sebesar 260.205 *bytes/sec* dan *threads 5000* sebesar 9.068.671 *bytes/sec*. Untuk mengetahui lebih rinci perolehan hasil pengujian pada kedua *web server* dapat dilihat pada tabel sebagai berikut:

Tabel 1. Kinerja *Web Server Nginx Bytes received per second*

Jumlah <i>Threads</i>	Waktu Uji	Paket Data yang Diterima (<i>bytes/sec</i>)
50	08:53:01	19.552
	08:53:02	41.548
100	09:07:55	34.216
	09:07:56	86.762
500	11:12:32	153.972
	11:12:33	356.824
	11.12.34	94.094
	11.12.47	218
	11.13.04	7.825
1000	09:24:39	113.646
	09:24:40	271.284
	09:24:41	355.602
	09:24:42	285.948
	09:24:43	193.076
5000	09:24:46	2.444
	09:29:40	105.092
	09:29:56	6.490.022

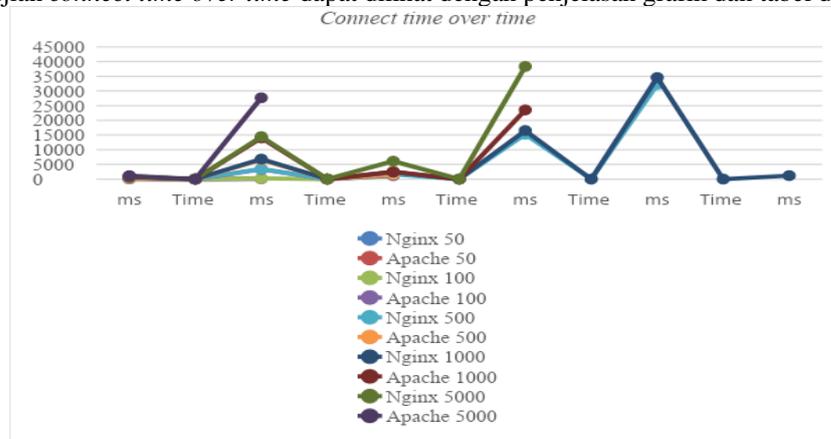
Tabel 2. Kinerja *Web Server Apache Bytes received per second*

Jumlah <i>Threads</i>	Waktu Uji	Paket Data yang Diterima (<i>bytes/sec</i>)
50	15:01:06	17.430
	15:01:07	44.840
100	14:56:49	84.660
	14:56:50	39.840
500	14:55:18	58.515
	14:55:19	206.670
	14:55:26	1.245
1000	14:52:09	12.450
	14:53:12	260.205
	14:53:44	8.603
5000	15:33:18	2.420
	15:33:33	9.068.671

Pada pengujian *nginx* dan *apache* pada tabel, dapat dilihat paket data yang diterima *web server apache* memiliki nilai tertinggi paket yang diterima yaitu sebesar 9.068.671 *bytes/sec* dengan jumlah *threads 5000*. Hasil yang bervariasi tersebut disebabkan karena menggambarkan kemampuan *nginx* dan *apache* dalam merespon *request*. Kemampuan server pada saat menerima *request* mempengaruhi banyaknya jumlah *response* yang dapat terkirim. Berdasarkan hasil tersebut maka dapat dikatakan pada saat pengujian *server* yang mampu menghasilkan total *byte* yang diterima dari *host* per detik paling banyak adalah *web server apache*.

3.2 Connect Time Over Time

Connect time over time menjelaskan waktu koneksi yang dibutuhkan *client* pada *server* dalam satuan *milisecond* (ms). Selanjutnya *server* akan segera merespon apa yang telah di *request* oleh *client* [3]. Berikut akan dijelaskan mengenai *connect time over time* dalam pengujian *web server nginx* dan *apache* dalam merespon *request* dari *client*. Hasil pengujian *connect time over time* dapat dilihat dengan penjelasan grafik dan tabel di bawah ini:



Gambar 3. Connect time over time

Pada grafik merepresentasikan parameter *Connect Time Over Time* secara garis besar *web server nginx* dengan *threads 50* memiliki nilai delay sebesar *111 ms*, *threads 100* sebesar *1.150 ms*, *threads 500* sebesar *32.185 ms*, *threads 1000* sebesar *2.381 ms* dan *threads 5000* sebesar *14.807 ms*. Sedangkan *web server apache* dengan *threads 50* memiliki nilai delay sebesar *46.1 ms*, *threads 100* sebesar *122 ms*, *threads 500* sebesar *3.077 ms*, *threads 1000* sebesar *7.195 ms* dan *threads 5000* sebesar *13.125 ms*. Hasil grafik lebih rinci dapat dilihat pada tabel 3:

Tabel 3. Kinerja Web Server Nginx Connect time over time

Jumlah Threads	Waktu Uji	Delay (ms)
50	08:53:01	47
	08:53:02	111
100	09:07:55	53
	09:07:56	78
	09:07:57	1150
500	11:12:32	99
	11:12:47	15.302
	11:13:04	32.185
1000	09:24:39	281
	09:24:43	2.381
	09:24:46	1.182
5000	09:29:40	225
	09:29:42	472
	09:29:54	3.561
	09:24:56	14.807

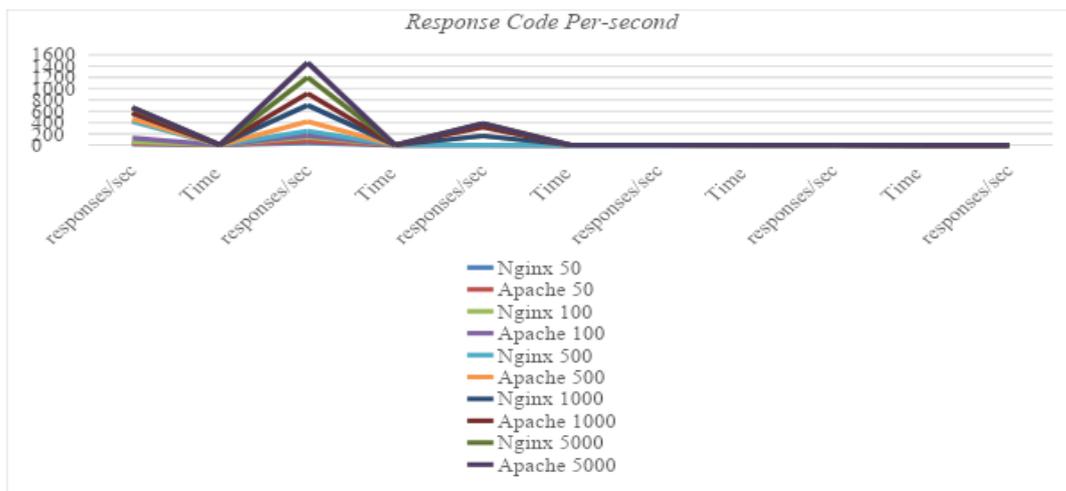
Tabel 4. Kinerja Web Server Apache Connect time over time

Jumlah Threads	Waktu Uji	Delay (ms)
50	15:01:06	46.1
	15:01:07	44.1
100	14:55:18	122
	14:55:24	44
500	14:55:18	122
	14:55:22	3.077
1000	14:53:10	109
	14:53:22	6.998
5000	15:33:18	183
	15:33:33	13.215

Pada pengujian tabel, hasil kinerja *web server nginx* dan *apache connect time over time* dengan jumlah *threads* yang sama, didapatkan hasil pada *web server nginx* memiliki hasil waktu koneksi yang dibutuhkan *client* pada *server* paling besar yaitu sebesar 32.185 ms dengan jumlah *threads* 500 dibanding *web server apache*. Hal ini disebabkan karena pada saat melakukan pengujian kondisi jaringan dipengaruhi oleh beberapa faktor, diantaranya banyaknya *hop*, kapasitas *bandwidth* dan padatnya *traffic* mempengaruhi kemampuan *web server* dalam mengirimkan *response* pada saat pengujian. Berdasarkan hasil tersebut didapatkan bahwa kemampuan *server* untuk mengirimkan *response* dengan *error rate* terkecil adalah *web server nginx*.

3.3 Response Code per Second

Response Code per Second menunjukkan kode response per detik yang dikembalikan selama waktu pengujian [3]. Berikut akan dijelaskan mengenai *response code per second* dalam pengujian *web server nginx* dan *apache*. Hasil pengujian tersebut dapat dilihat dengan penjelasan grafik dan tabel di bawah ini:



Gambar 4. *Response Code Per-second*

Pada grafik dapat dilihat hasil dari parameter *response code per second* secara garis besar hasil *web server nginx* sanggup merespon data paling banyak dengan *threads* 50 sebesar 34.00 *responses/sec*, *threads* 100 sebesar 71.00 *responses/sec*, *threads* 500 sebesar 292.00 *responses/sec*, *threads* 1000 sebesar 291.00 *responses/sec*, dan *threads* 5000 sebesar 285.00 *responses/sec*. Sedangkan pada *web server apache* sanggup merespon data paling banyak dengan *threads* 50 sebesar 36.00 *responses/sec*, *threads* 100 sebesar 68.00 *responses/sec*, *threads* 500 sebesar 166.00 *responses/sec*, *threads* 1000 sebesar 209.00 *responses/sec*, dan *threads* 5000 sebesar 260.00 *responses/sec*. Secara rinci hasil pengujian *response code per second* *web server nginx* dan *apache* dapat dilihat pada tabel berikut ini:

Tabel 5. Kinerja Web Server Nginx Response Code Per-second

Jumlah Threads	Waktu Uji	Waktu Respon Data (responses/sec)
50	08:53:01	16.00
	08:53:02	34.00
100	09:07:55	28.00
	09:07:56	71.00
	09:07:57	1.00
500	11:12:32	126.00
	11:12:33	292.00
	11:12:34	77.00
	11:12:47	1.00
1000	09:24:39	98.00
	09:24:41	291.00
	09:24:46	2.00
5000	09:29:40	86.00
	09:29:46	285.00
	09:29:56	67.00

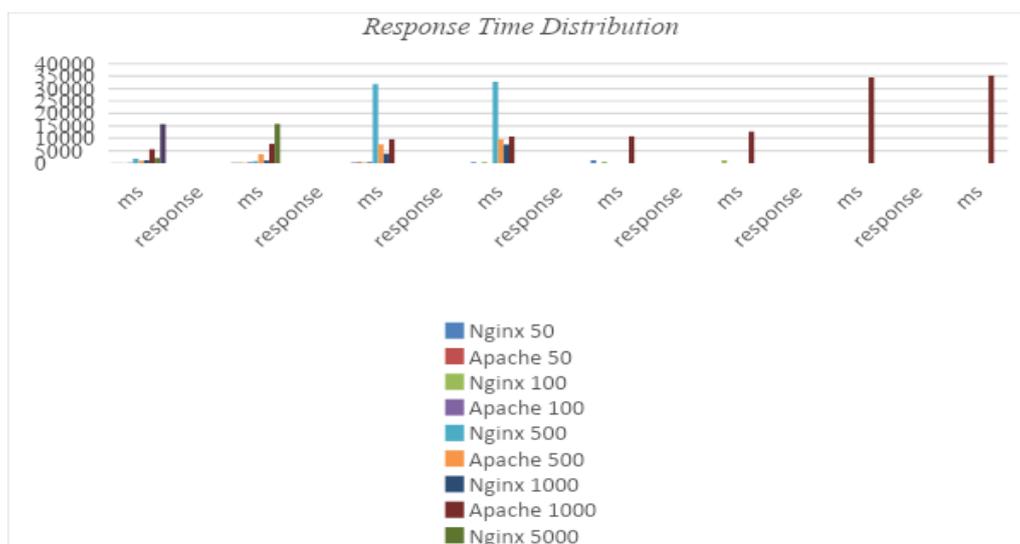
Tabel 6. Kinerja *Web Server Apache Response Code Per-second*

Jumlah Threads	Waktu Uji	Waktu Respon Data (<i>responses/sec</i>)
50	15:01:06	14.00
	15:01:07	36.00
100	14:56:49	68.00
	14:56:50	32.00
500	14:55:18	47.00
	14:55:19	166.00
	14:55:20	160.00
	14:55:28	1.00
1000	14:53:10	10.00
	14:53:12	209.00
	14:53:13	152.00
	14:53:14	1.00
	14:53:44	2.00
5000	15:33:18	11.00
	15:33:30	260.00

Pada pengujian tabel diatas, kinerja *web server nginx* dan *apache response code per-second* dengan jumlah *threads* yang sama, dari kedua *web server* didapatkan *response* per detik yang dikembalikan selama waktu pengujian hasil terbesar pada *web server nginx* 292.00 *responses/sec* dengan jumlah *threads* 500. Hasil yang variatif tersebut disebabkan karena pada saat pengujian kemampuan *nginx* dan *apache* untuk mengirimkan status *header* pada *response* pada saat membalas *request*. Hal ini bergantung pada kemampuan *web server* dalam mengelola *request* yang masuk dan membalas *request* tersebut. Berdasarkan hasil tersebut maka pengukuran kinerja dari infrastruktur jaringan *server* juga memberikan *web server apache* yang lebih baik dibanding *web server nginx* dikarenakan *web server apache* memiliki *delay response* yang lebih kecil dibanding *web server nginx*, sementara *web server apache* juga dapat memberikan *response* yang lebih banyak dalam kurun waktu yang lebih singkat dibandingkan *web server nginx*.

3.4 Response Time Distribution

Response time distribution menyajikan distribusi waktu *response* selama pengujian dilakukan. Sumbu X menampilkan waktu *response* yang dikelompokkan berdasarkan interval, sedangkan sumbu Y menampilkan jumlah sampel yang terdapat pada setiap interval [3]. Berikut akan dijelaskan mengenai *response time distribution* dalam pengujian *web server nginx* dan *apache*. Hasil pengujian tersebut dapat dilihat dengan penjelasan grafik dan tabel di bawah ini:



Gambar 5. *Response Time Distribution*

Hasil yang ditampilkan pada grafik adalah hasil dari pengujian *Response Time Distribution* secara garis besar *web server nginx* merespon data yang diberikan *client* tertinggi dengan jumlah *threads* 50 sebesar 24 data,

threads 100 sebesar 62 data, *threads* 500 sebesar 186 data, *threads* 1000 sebesar 314 data dan *threads* 5000 sebesar 347 data. Sedangkan *web server apache* merespon data yang diberikan *client* tertinggi dengan jumlah *threads* 50 sebesar 36 data, *threads* 100 sebesar 93 data, *threads* 500 sebesar 78 data, *threads* 1000 sebesar 40 data, dan *threads* 5000 sebesar 532 data. Untuk mengetahui lebih jelas lagi mengenai hasil pengujian dapat dilihat pada tabel berikut:

Tabel 7. Kinerja *Web Server Nginx Response Time Distribution*

Jumlah <i>Threads</i>	Durasi Data (ms)	Jumlah Data
50	0-100	8
	100-200	24
	200-300	15
	300-1200	1
100	0-100	6
	100-200	62
	200-400	24
	400-1100	5
	1100-1200	1
500	0-1800	496
	700-800	186
	1700-31800	1
	31800-32700	4
1000	0-1200	314
	1000-1100	98
	3700-7400	1
	7400-7500	2
5000	2000-15100	1
	15700-15800	347

Tabel 8. Kinerja *Web Server Apache Response Time Distribution*

Jumlah <i>Threads</i>	Durasi Data (ms)	Jumlah Data
50	0-100	11
	100-300	36
	300-400	3
100	100-300	93
	300-400	3
	400-500	4
500	900-1000	78
	3500-9600	1
1000	5500-7300	1
	7300-9500	40
	9600-10600	1
	10600-10700	15
	10700-12000	1
	12000-34400	28
	34400-35000	1
35000-35200	3	
5000	14300-15700	532

Pada pengujian tabel hasil kedua *web server nginx* dan *apache response time distribution* dengan jumlah *threads* yang sama. Jumlah sampel yang terdapat pada setiap interval tertinggi pada pengujian *web server apache* sebanyak 532 data dengan durasi 14300-15700 ms jumlah *threads* 5000. Hasil pengujian tersebut disebabkan oleh beberapa faktor yang mempengaruhi, diantaranya adalah banyaknya *hop*, kapasitas *bandwidth* dan padatnya *traffic* mempengaruhi kemampuan *web server* dalam mengirimkan *response* pada saat pengujian. Berdasarkan pengujian pada tabel dapat dikatakan bahwa jumlah data paling banyak pada *web server apache* dalam menyajikan distribusi waktu *response* selama pengujian

3.5 Latency

Delay atau *latency* merupakan waktu yang dibutuhkan dalam sebuah pengiriman dari perangkat komputer satu dengan yang lainnya. Proses *latency* dalam sebuah transmisi jaringan komputer disebabkan adanya antrian panjang atau menghindari jalur untuk menghindari kemacetan. Untuk menghitung nilai *latency* pada proses transmisi yaitu membagi antara panjang paket (bit) dibagi dengan link bandwidth (bit/s) [4]. Pada server *Nginx* dilakukan perhitungan *latency* berdasarkan total jumlah delay dibagi dengan total paket data diterima [5] yang dapat dijelaskan sebagai berikut:

- a. 50 Threads: rata-rata $delay = \frac{9.184}{50} = 183,68$, *latency* 50 Threads dapat dikategorikan baik.
- b. 100 Threads: rata-rata $delay = \frac{19.536}{100} = 195,36$, *latency* 100 Threads dapat dikategorikan baik.
- c. 500 Threads: rata-rata $delay = \frac{544.458}{500} = 1.088,91$, *latency* 500 Threads dapat dikategorikan buruk.
- d. 1000 Threads: rata-rata $delay = \frac{2.123.39}{1000} = 2.123,39$, *latency* 1000 Threads dapat dikategorikan buruk.
- e. 5000 Threads: rata-rata $delay = \frac{60.504.850}{5000} = 12.100,97$, *latency* 5000 Threads dapat dikategorikan buruk.

Pada server *Apache* dilakukan perhitungan *latency* berdasarkan total jumlah delay dibagi dengan total paket data diterima [5] yang dapat dijelaskan sebagai berikut:

- a. 50 Threads: rata-rata $delay = \frac{6.493}{50} = 129,86$, *latency* 50 Threads dapat dikategorikan baik.
- b. 100 Threads: rata-rata $delay = \frac{15.222}{100} = 152,22$, *latency* 100 Threads dapat dikategorikan baik.
- c. 500 Threads: rata-rata $delay = \frac{818.743}{500} = 1.637,48$, *latency* 500 Threads dapat dikategorikan buruk.
- d. 1000 Threads: rata-rata $delay = \frac{4.477.602}{1000} = 4.477,60$, *latency* 1000 Threads dapat dikategorikan buruk.
- e. 5000 Threads: rata-rata $delay = \frac{70.616.381}{5000} = 14.123,28$, *latency* 5000 Threads dapat dikategorikan buruk.

Adapun rangkuman perhitungan *latency* pada komparasi kinerja *web server Nginx* dan *Apache* dapat dijelaskan pada tabel berikut ini.

Tabel 9. Komparasi Kinerja *Nginx* dan *Apache*

Web Server	Samples	Latency				Persentase Error %
		Average	Median	Min	Max	
<i>Nginx</i>	50	183	137	95	1144	0.00
	100	195	160	91	1109	0.00
	500	1088	746	191	32652	0.00
	1000	2123	1851	513	7498	0.00
	5000	12100	15526	535	16641	59.96
<i>Apache</i>	50	129	108	92	397	0.00
	100	152	141	100	454	0.00
	500	1637	1439	200	9596	0.00
	1000	4477	3592	323	35122	0.20
	5000	14123	14817	629	15633	79.54

Berdasarkan standar *delay TIPHON* untuk *web server nginx*, dihitung jumlah *delay* dan jumlah paket yang diterima oleh *client*, untuk parameter 50 dan 100 *threads* hasil *latency* dapat dikatakan baik dengan nilai di antara 150 - 300 ms, selanjutnya untuk parameter 500, 1000 dan 5000 *threads*, hasil *latency* dapat dimasukkan ke dalam kategori buruk dengan nilai lebih dari 450 ms. Berdasarkan standar *delay TIPHON* untuk *web server apache*, dihitung jumlah *delay* dan jumlah paket yang diterima oleh *client*, untuk parameter 50 *threads*, hasil *latency* dapat dimasukkan kategori sangat baik dengan nilai *latency* kurang dari 50 ms, untuk 100 *threads* hasil *latency* dapat dikatakan baik dengan nilai di antara 150 - 300 ms, selanjutnya untuk parameter 500, 1000 dan 5000 *threads*, hasil *latency* dapat dimasukkan ke dalam kategori buruk dengan nilai lebih dari 450 ms. Hasil *output* dari *jmeter* beragam disebabkan oleh karakteristik dari masing - masing *web server*, perbedaan mendasar adalah bagaimana cara keduanya menangani *request*, *Apache* menyediakan berbagai modul multi-pemrosesan untuk menangani permintaan klien dan lalu lintas web, *nginx* dirancang untuk menangani beberapa permintaan klien secara bersamaan dengan sumber daya perangkat keras minimal.

4. KESIMPULAN

Berdasarkan hasil pengujian dari lima parameter tersebut dapat ditarik kesimpulan bahwa kemampuan *server* untuk mengirimkan *response* dengan jumlah *byte* yang paling besar adalah menggunakan *web server apache*, sedangkan kemampuan *server* untuk mengirimkan *response* dengan *error rate* terkecil adalah *web server nginx*. Pengukuran kinerja dari infrastruktur jaringan server juga memberikan *web server apache* yang lebih baik dibanding *web server nginx* dikarenakan *web server apache* memiliki *delay response* yang lebih kecil dibanding *web server nginx*, sementara *web server apache* juga dapat memberikan *response* yang lebih banyak dalam kurun waktu yang

lebih singkat dibandingkan *web server nginx*. Maka dapat disimpulkan bahwa *web server apache* memiliki hasil lebih efektif dalam kinerja *web service* terhadap integrasi pada multi basis data.

UCAPAN TERIMA KASIH

Ucapan terima kasih dapat diperuntukkan kepada pihak-pihak yang telah membantu dalam penelitian, instansi yang menjadi objek penelitian, bisa juga kepada pihak yang membantu dalam publikasi artikel.

DAFTAR PUSTAKA

- [1] Alisjabbana, I. (1980). Teknologi dan Perkembangan. Jakarta : Yayasan Idayu. <https://opac.perpusnas.go.id/DetailOpac.aspx?id=431718>
- [2] Sugiyono, P. D. (2019). Metode Penelitian Kuantitatif Kualitatif dan R&D (Cetakan ke 1 ed.). Alfabeta. ISBN 978-602-289-533-6
- [3] Hasbullah, Mochamad Sufiandi. 2018. Analisis Perbandingan Web Server Apache dan Nginx menggunakan Apache Jmeter pada Server Aplikasi AIS yang Berbasis Clustering Server. Universitas Islam Negeri Syarif Hidayatullah Jakarta. Jakarta. (diakses pada tanggal 8 Desember 2021)
- [4] Lorenz, D. H., & Orda, A. (2002, Agustus 7). Optimal Partition of Qos Requirements on Unicast Paths and Multicast Trees. Optimal Partition of Qos Requirements on Unicast Paths and Multicast Trees, 10(1), 102-114. DOI: 10.1109/90.986559
- [5] Irawati, Indrarini Dyah, dkk. Jaringan Komputer dan Data Lanjut. Yogyakarta: Deepublish, 2015
- [6] Yogiswara, Wijono, dan Harry Soekotjo Dahlan. Kinerja Web Service pada Proses Integrasi Data, 2014
- [7] Albert Yakobus Chandra. Analisis Performansi Antara Apache & Nginx Web Server dalam Menangani Client Request, 2019
- [8] Andri Jiwandono. Analisa Perbandingan Kinerja Web Server Apache, Nginx, dan Litespeed Dengan Menggunakan Metode Stress Test. 2021
- [9] Cahyanto, Triawan Adi. "Penerapan Teknologi Web Service Pada Sistem Informasi Data Rekam Medis Rumah Sakit XYZ." Query: Journal of Information Systems 2.1 (2018).
- [10] Yogiswara. Kinerja Web Service pada Web Server Apache, Ngin-X dan IIS-7. 2015